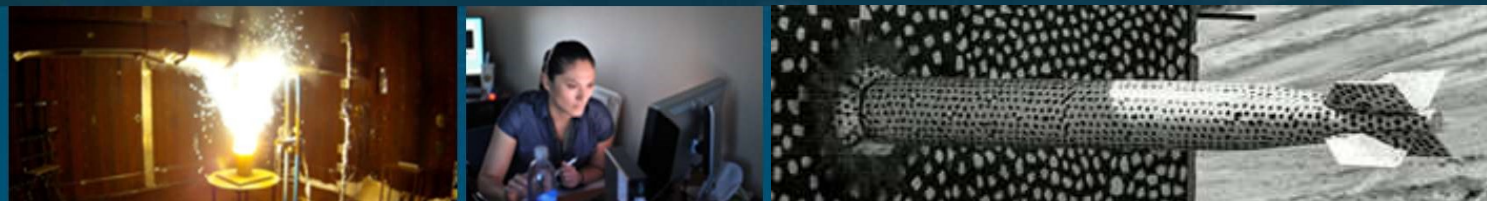


University of Massachusetts, Amherst
Department of Electrical and Computer Engineering
ECE Graduate Seminar



The Reversible Computing Future



Wednesday, October 20th, 2021

Michael P. Frank, Center for Computing Research

with collaborators: Robert Brocato, Rupert Lewis, Nancy Missert & Brian Tierney (Sandia), Kevin Osborn & Lingqi Yu (LPS), Erik DeBenedictis (Zettaflops, LLC), Karpur Shukla (Brown), Rudro Biswas, Dewan Woods & Rishabh Khare (Purdue), Tom Conte & Anirudh Jain (Georgia Tech).

Approved for public release, SAND2021-13154 PE



Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.



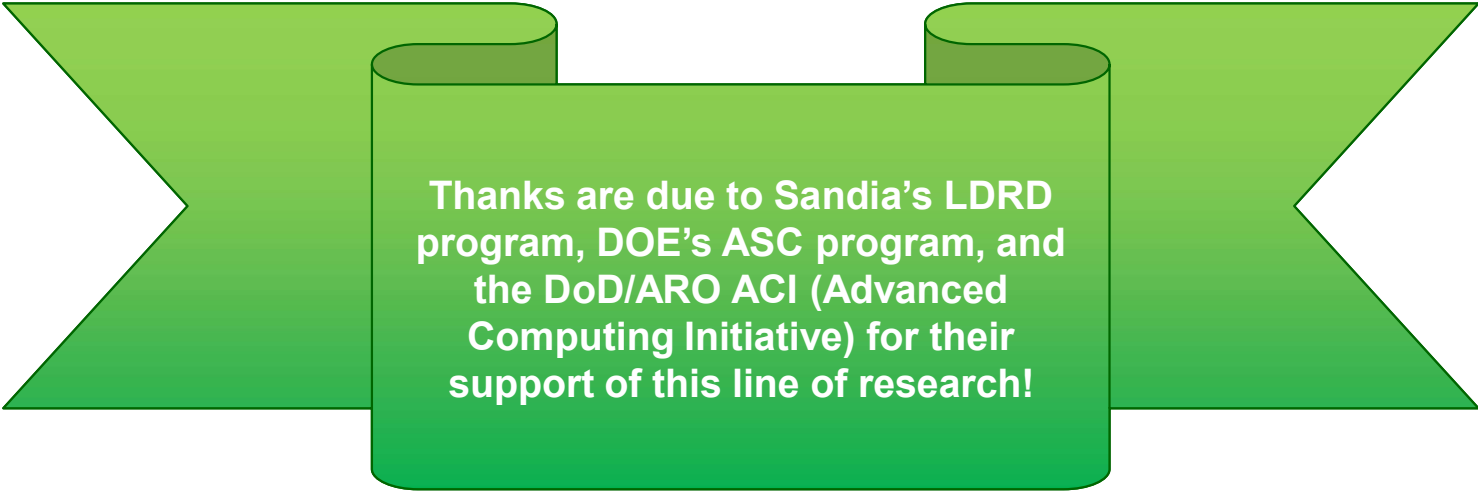
The Reversible Computing Future

The cost efficiency of computing has been improving at an exponential rate over the last seventy years, which has led to a vast proliferation of computing applications throughout the economy, and in our daily lives. Can this trend continue? One can show, from basic statistical physics and information theory, that the conventional *non-reversible* paradigm for digital computing, which relies on primitive operations that discard information, will soon run up against fundamental limits on its energy efficiency, and therefore cost efficiency. The only way to circumvent these limits, in digital computing, is to migrate to the *reversible* computing paradigm, which restructures digital circuits in ways that avoid information loss. In principle, there is no limit to the energy efficiency and cost efficiency that reversible computing can potentially attain, as the technology improves. However, developing this technology will ultimately require major changes at all levels of the computing technology stack, from devices to systems. We review existing implementation concepts for reversible computing based on CMOS and superconducting technologies, and outline the major physics and engineering challenges that will need to be addressed in order for the field to move forward.

Contributors to the larger effort

- Full group at Sandia:

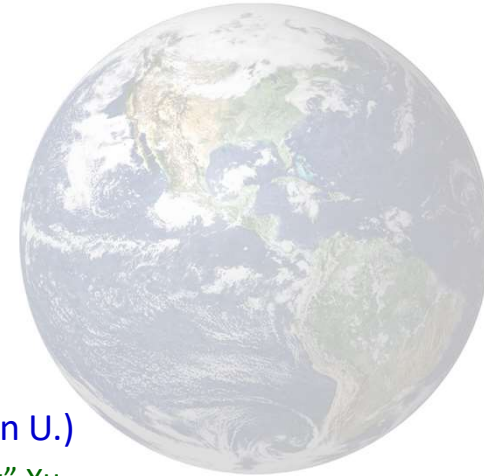
- Michael Frank (Cognitive & Emerging Computing)
- Robert Brocato (RF MicroSystems)
- David Henry (MESA Hetero-Integration)
- Rupert Lewis (Quantum Phenomena)
- Nancy Missert (Nanoscale Sciences)
 - Matt Wolak (now at Northrop-Grumman)
- Brian Tierney (Rad Hard CMOS Technology)



Thanks are due to Sandia's LDRD program, DOE's ASC program, and the DoD/ARO ACI (Advanced Computing Initiative) for their support of this line of research!

- Thanks are also due to the following colleagues & external collaborators:

- Erik DeBenedictis
- Kevin Osborn (LPS/JQI)
 - Liuqi Yu
- Steve Kaplan
- Rudro Biswas (Purdue)
 - Dewan Woods
 - Rishabh Khare
- Karpur Shukla (CMU/Brown U.)
 - w. Prof. Jingming "Jimmy" Xu
 - Also w. Victor Albert (CalTech)
- Tom Conte (Georgia Tech/CRNCH)
 - Anirudh Jain
- David Guéry-Odelin (Toulouse U.)
- FAMU-FSU College of Engineering:
 - Sastry Pamidi (ECE Chair)
 - Jerris Hooker (Instructor)
 - 2019-20 students:
 - Frank Allen, Oscar L. Corces, James Hardy, Fadi Matloob
 - 2020-21 students:
 - Marshal Nachreiner, Samuel Perlman, Donovan Sharp, Jesus Sosa



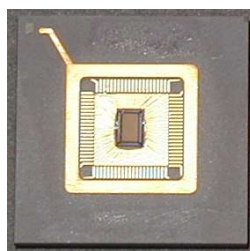
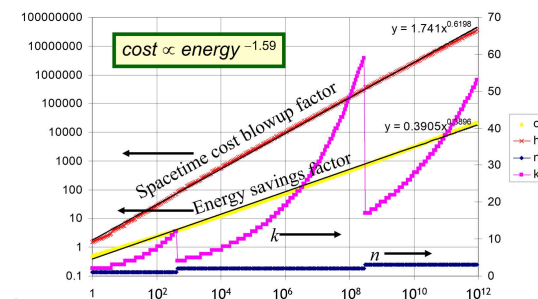
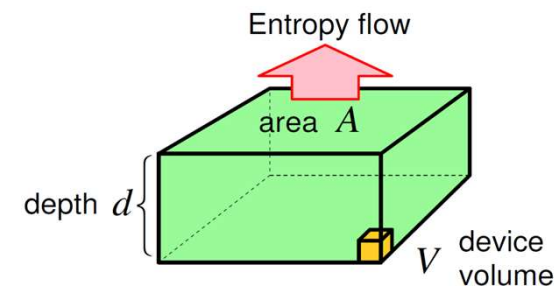
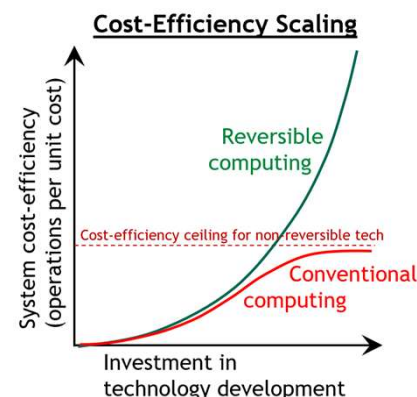
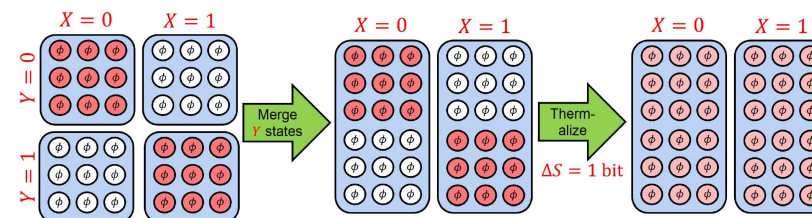
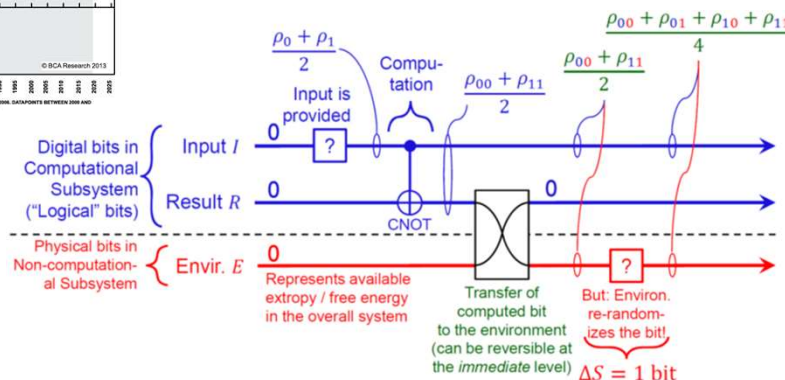
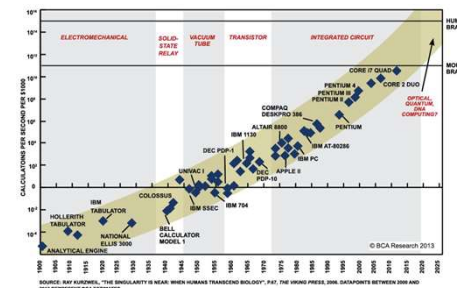


Talk Abstract/Outline

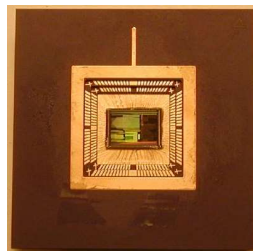


The Reversible Computing Future

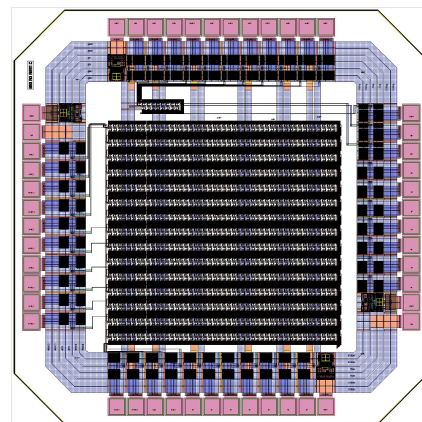
- Computing *efficiency* has been increasing exponentially for decades...
 - Can this trend continue into the future? How much farther?
- The energy efficiency of the usual *non-reversible* digital paradigm is limited...
 - Due to fundamental limits implied by basic statistical physics and information theory.
- Avoiding these limits in digital machines requires *reversible computing* (RC)...
 - Means, computing in a way that eliminates (or at least reduces) local information loss.
 - We know of no fundamental limit to the possible energy (& cost) efficiency of RC.
 - But, RC requires major changes to the technology stack (eventually at all levels).
- Emphasis of today's talk:
 - Review existing CMOS & superconducting implementation technologies for RC.
 - Outline the major outstanding physics & engineering challenges to move forward.



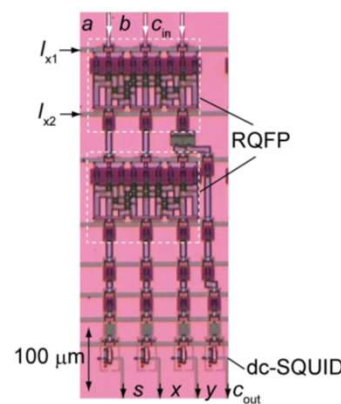
FlatTop
(MIT '96)



Pendulum
(MIT '99)



2LAL Shift Register (Sandia '20)



RQFP Full Adder (Yokohama '18)

Trend of Improving Cost-Efficiency of Computing

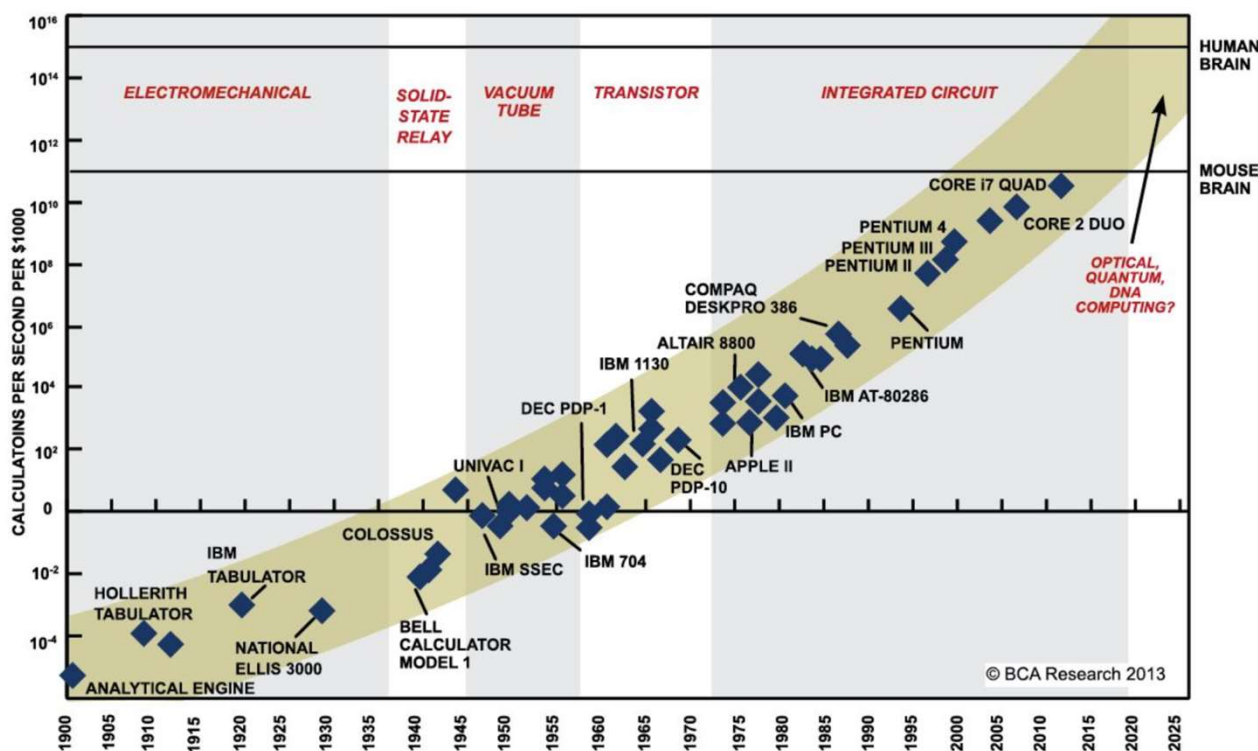
$$\eta_c = \frac{\#ops}{Cost} = \frac{1}{C_{op}}$$



Since at least 1950 (and really even longer), the *cost-efficiency* η_c of computing has improved exponentially...

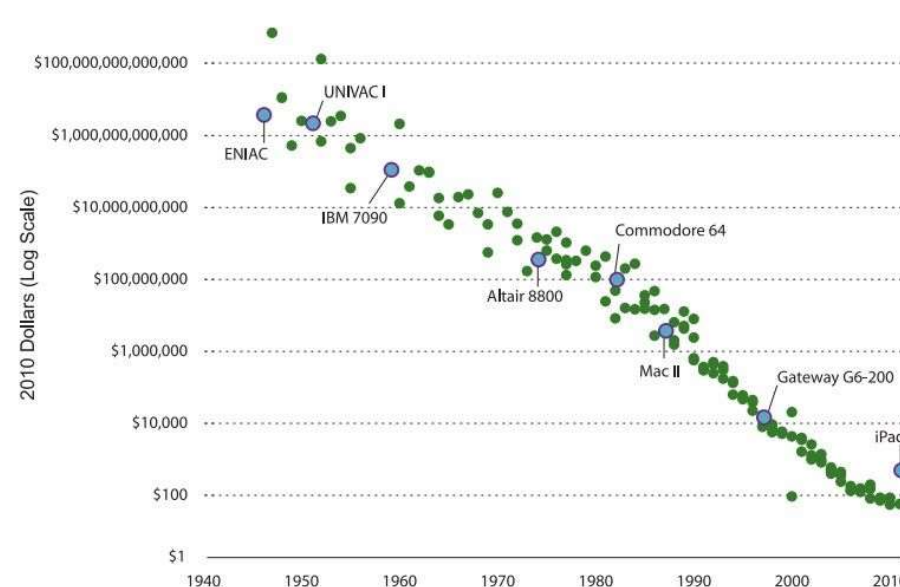
- Can generically define cost-efficiency in terms of *computational operations performed (e.g., FLOPs) per dollar spent*.
 - Maximizing cost-efficiency equates to minimizing the cost to perform (some given number of) operations over the system's lifetime.
 - In general, this includes both costs to manufacture/deploy the system, and the lifetime cost of operating the system (including energy costs).
- In typical contexts today, the practical lifetime L of most computing systems is relatively fixed (a few years, say).
 - And also, for most applications, there is a maximum tolerable latency ℓ until the result of a given computational task must be obtained.
- So, generally we care about not *just* maximizing η_c , but also minimizing cost/op for operations within some fixed timeframe,
 - Which translates to increasing both *performance per unit (manufacturing) cost*, as well as (accounting for energy costs) *performance per unit power dissipation*.

$$C_{tot} = C_{mfg} + C_{oper}$$



SOURCE: RAY KURZWEIL, "THE SINGULARITY IS NEAR: WHEN HUMANS TRANSCEND BIOLOGY", P.67, THE VIKING PRESS, 2006. DATAPPOINTS BETWEEN 2000 AND 2012 REPRESENT BCA ESTIMATES.

Cost of Computing Power Equal to an iPad 2



Note: The iPad2 has computing power equal to 1600 million instructions per second (MIPS). Each data point represents the cost of 1600 MIPS of computing power based on the power and price of a specific computing device released that year.

Source: Moravec n.d..

Semiconductor Roadmap is Ending...

Thermal noise on gate electrodes of minimum-width segments of FET gates leads to significant channel PES fluctuations if $E_g \lesssim 1\text{-}2\text{ eV}$!

- This increases leakage, impairs practical device performance
- Thus, roadmap has minimum gate energy asymptoting to $\sim 2\text{ eV}$

Further, real logic circuits incur many *compounding* overhead factors *multiplying* this raw transistor-level limit:

- Transistor width $10\text{-}20\times$ minimum width for fastest logic.
- Parasitic (junction, etc.) transistor capacitances ($\sim 2\times$).
- Multiple (~ 2) transistors fed by each input to a given logic gate.
- Fan-out of each gate to a few (~ 3) downstream logic gates.
- Parasitic wire capacitance ($\sim 2\times$).

Due to all these overhead factors, the energy of each logic bit in real logic circuits is necessarily many times larger than the minimum-width gate energy!

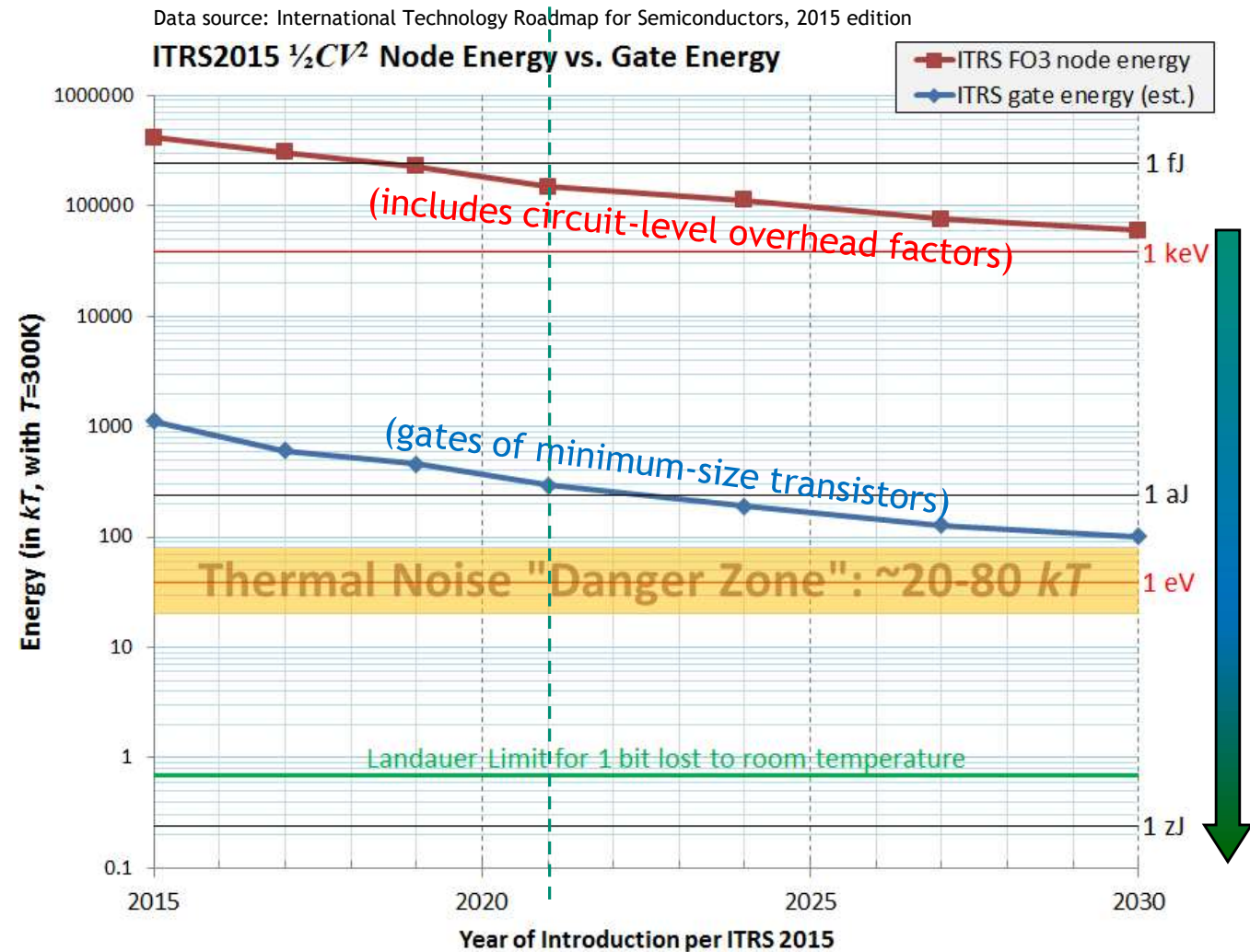
- $375\text{-}600\times$ (!) larger in ITRS'15.
- \therefore Practical bit energy for irreversible CMOS logic asymptotes to $\sim 1\text{ keV}$!

Practical, real-world logic circuit designs can't just magically cross this $\sim 500\times$ architectural gap!

- \therefore Thermodynamic limits imply much larger practical limits!
- The end is near!

This is Now!

Only about a decade left...



Only reversible computing can take us from $\sim 1\text{ keV}$ at the end of the CMOS roadmap, all the way down to $\ll kT$.

Basic Reversible Computing Theory

(For full proofs, see [arxiv.org:1806.10183](https://arxiv.org/1806.10183))

Fundamental theorem of traditional reversible computing:

- A deterministic computational operation is (unconditionally) non-entropy-ejecting if and only if it is *unconditionally* logically reversible (injective over its entire domain).

Fundamental theorem of generalized reversible computing:

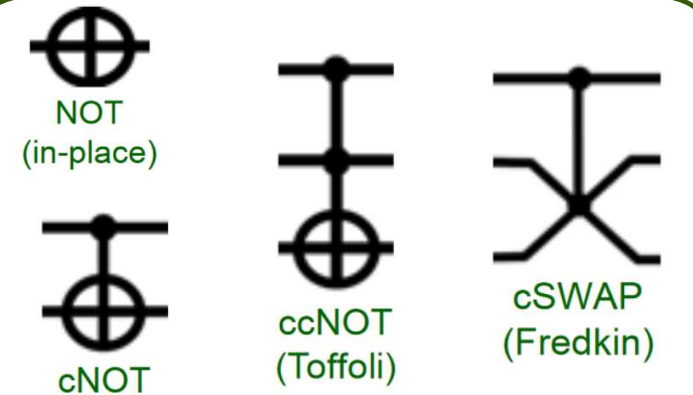
- A *specific* (contextualized) deterministic computational process is (specifically) non-entropy-ejecting if and only if it is *specifically* logically reversible (injective over the set of *nonzero-probability* initial states).
- Also, for any deterministic computational operation, which is conditionally reversible under some assumed precondition, then the entropy required to be ejected by that operation approaches 0 as the probability that the precondition is satisfied approaches 1.

Bottom line: To avoid requiring Landauer costs, it is *sufficient to just have reversibility when some specified preconditions are satisfied*.

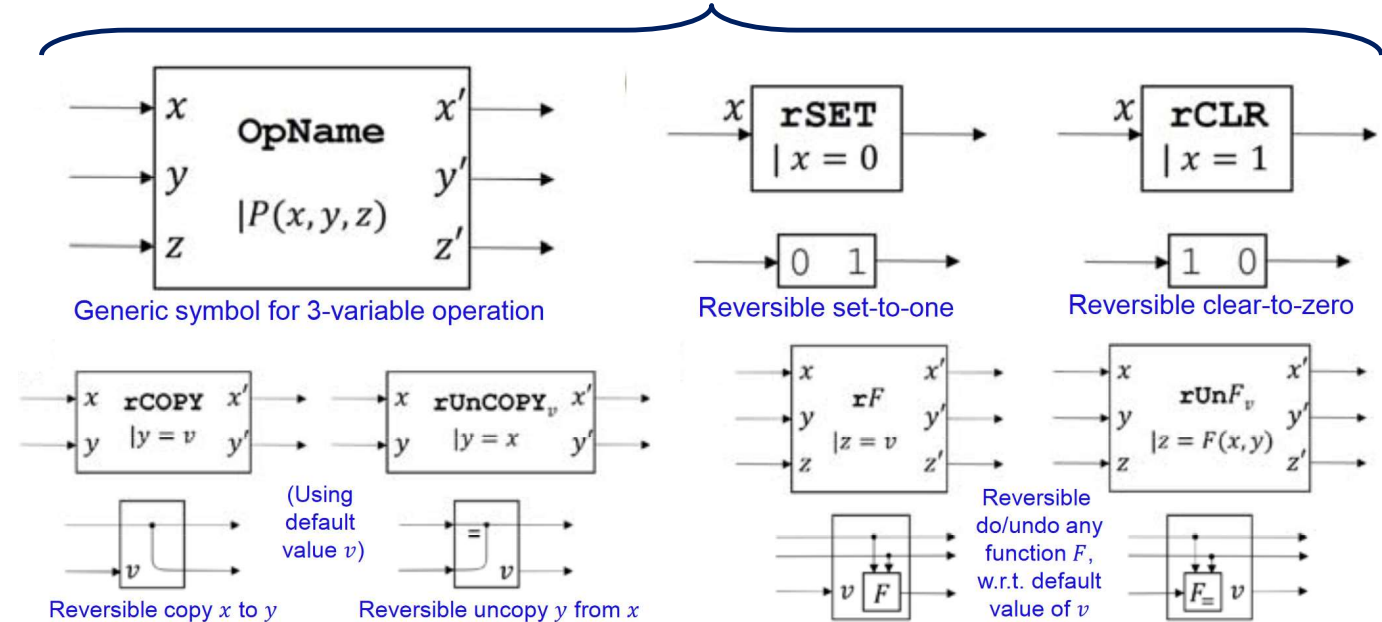
- Basis for practical engineering implementations.
- Exemplified by Adiabatic CMOS.



Traditional *Unconditionally* Reversible “Gates” (Operations)



Generalized *Conditionally* Reversible Operations



9 Why Reversible Computing Wins Despite Its Overheads!

$$\eta_c = \frac{\#ops}{Cost}$$



Bumper-sticker slogan: “*Running Faster by Running Slower!*” (Wait, what?) More precisely:

- Reversible technology is so energy-efficient that we can overcome its overheads (including longer transition times!) by using much greater parallelism to increase aggregate performance within system power constraints.
- This is borne out by a detailed economic/systems-engineering analysis.

Bottom line: The computational *performance per unit budgetary cost* on parallelizable computing workloads can become as large as desired, given only that *both terms* in this expression for total *cost per operation* C_{op} can be made sufficiently small:

$$C_{op} = c_E \cdot E_{diss,op} + c_M(s_{elem} \cdot t_{delay}).$$

where:

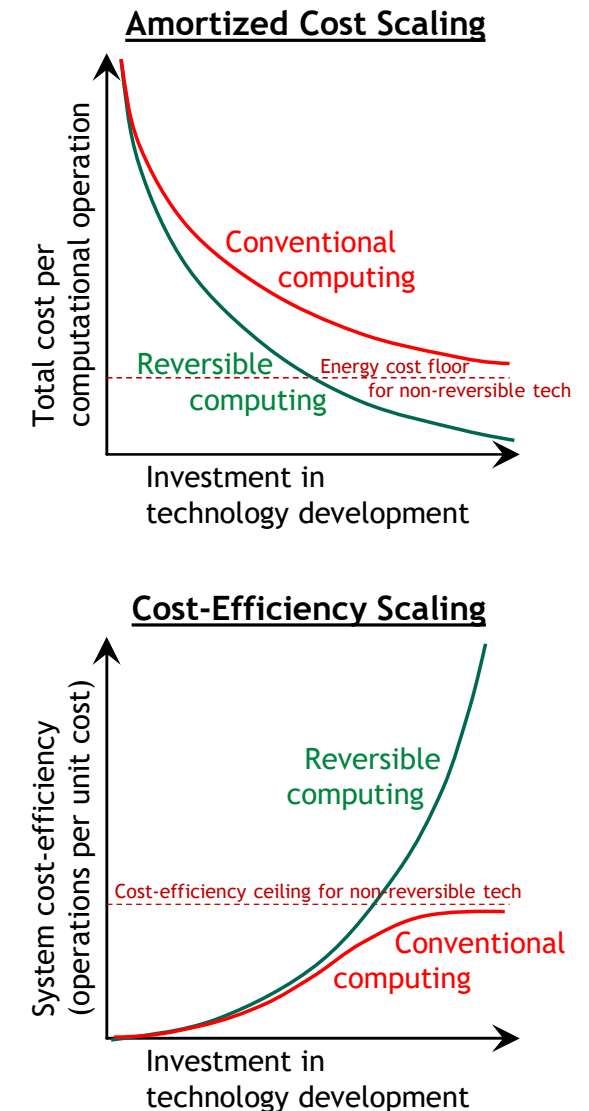
- c_E is the operating cost C_{oper} attributable to supplying power/cooling, divided by energy delivered.
- $E_{diss,op}$ is the system energy dissipation, divided by number of operations performed.
- c_M is the total cost C_{mfg} for system manufacturing & installation, *divided by* the number n_{elem} and physical size s_{elem} (in appropriate units) of individual computing elements, & the system’s total useful lifetime t_{life} .
- t_{delay} is the average time delay between instances of re-use of each individual computing element.

Two key observations:

- The cost per operation of *all* conventional computing *approaches a hard floor* due to Landauer.
 - Assuming *only* that the economic cost of operation *per Joule delivered* cannot become arbitrarily small.
- But, there is no clear barrier to making the manufacturing cost coefficient c_M *ever smaller* as manufacturing processes are refined (and/or the deployed lifetime of the system increases).

\therefore Nothing prevents system-level cost efficiency of reversible machines from becoming *arbitrarily* larger than conventional ones, *even* if we have to scale t_{delay} and/or s_{elem} up as we scale $E_{diss,op}$ down!

$$C_{tot} = C_{mfg} + C_{oper}$$



Performance Per-Area Scaling with Machine Thickness



Frank & Knight 1997, doi:[10.1088/0957-4484/9/3/005](https://doi.org/10.1088/0957-4484/9/3/005)

Assumptions of this simple analysis include:

- Classic adiabatic ($E_{\text{diss,op}} \propto 1/t$) scaling.
- Fixed operating temperature.
- Constant volume and mass per device.
- Bounded entropy flux density F_S .
- No algorithmic overheads for reversibility.

Upshot: Sustained performance of reversible machines asymptotically scales as $A\sqrt{d}$, which is $\sqrt{d} \times$ better than scaling of irreversible machines.

- Here, A is the area of the machine's minimal bounding surface, and d is the *depth* or thickness of the machine (along its thinnest dimension).

More detailed analyses also account for the impact of considering the algorithmic overheads of reversibility.

- Spoiler: Reversible computing still wins!

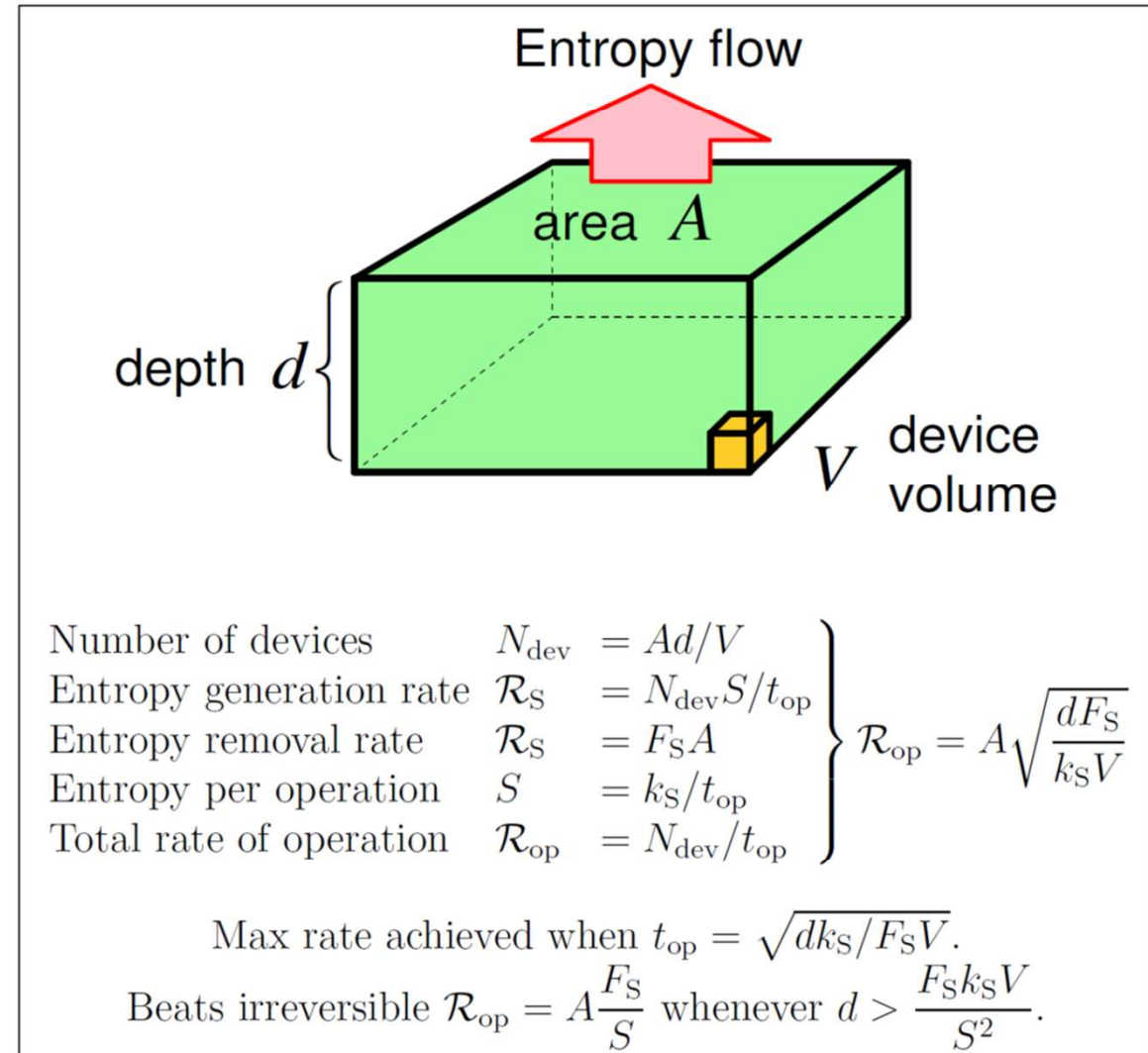


Figure 6.1: Speed limit for reversible machines of minimum-surface area $\Theta(A)$ and thickness $d \lesssim A^{1/2}$. The maximum rate of computation scales as $\Theta(A\sqrt{d})$.

Accounting for Nonidealities

Earlier analyses assumed that leakage can be engineered to be as small as necessary for it not to be limiting (which may be an OK assumption for *some* technologies) and negligible algorithmic overheads (which may be an OK assumption for *some* problems).

- But, can we still show an advantage even when making more pessimistic/realistic assumptions?
 - Answer is yes!

Even for worst-case problems, we can always at least still use the “Frank ‘02” algorithm (Bennett ‘89 variant).

- And, even better general “reversiblization” algorithms may yet be discovered in the future.

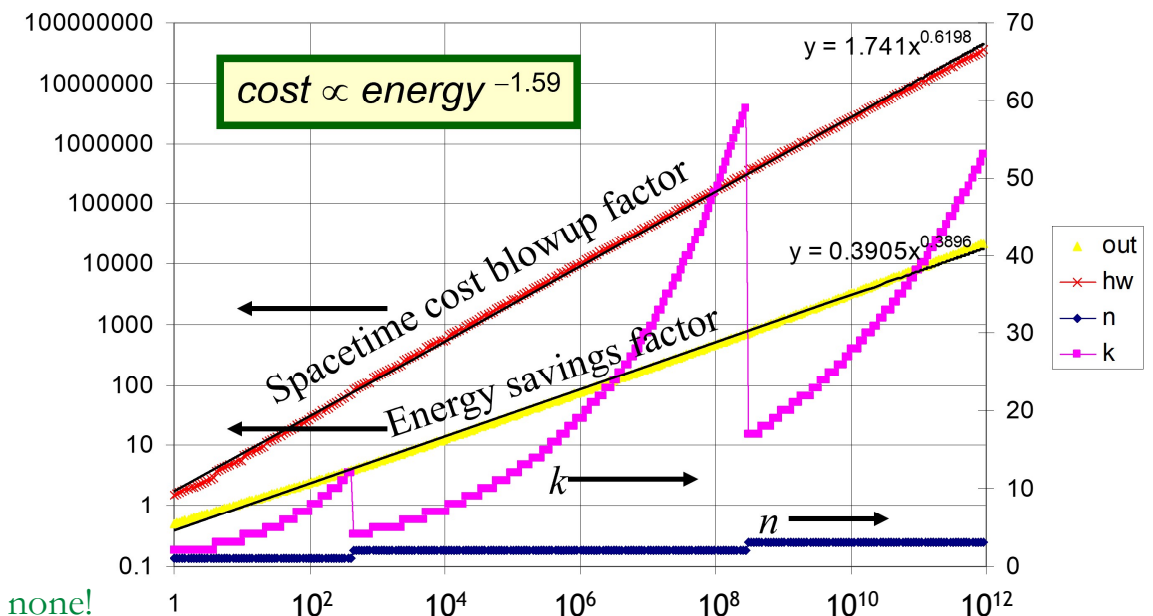
Then, as the technology is improved, and leakage is reduced, we can adjust the parameters of the algorithm to minimize the total cost

- Including both energy and spacetime/mfg. associated costs.

We find that we can reduce total lifetime *system* cost by any factor of N if we just reduce leakage by $\sim N^{2.56}$ and time-amortized per-device manufacturing cost by $\sim N^{1.59}$.

- Example: To achieve an $N = 1,000 \times$ overall efficiency boost, reduce leakage by $47.8M \times$ and mfg. cost/device by $59,000 \times$.
 - Ambitious but doable!! This gives us a way forward, where otherwise there is none!

Worst-Case Energy/Cost Tradeoff (Optimized Bennett-89 Variant)





Reversible Computing Technologies in Semiconducting Platforms

The Reversible Computing Future

Adiabatic Circuits in CMOS: A Brief History

A selection of some early papers:

Fredkin and Toffoli (MIT), 1978 (DOI:10.1007/978-1-4471-0129-1_2)

- Unfinished circuit concept based on idealized capacitors and inductors
 - How to control switches to do logic was left unspecified
 - Large design overhead—Roughly one inductor per gate

Seitz *et al.* (CalTech), 1985 (CaltechCSTR:1985.5177-tr-85)

- Realistic MOSFET switches; more compact integration (off-chip L)
- Not yet known to be general-purpose; required careful tuning

Koller and Athas (USC/ISI), 1992 (DOI:10.1109/PHYCMP.1992.615554)

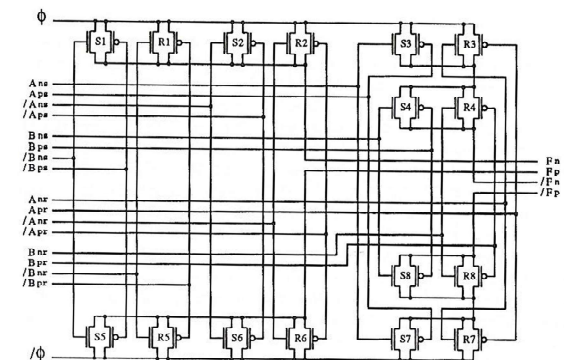
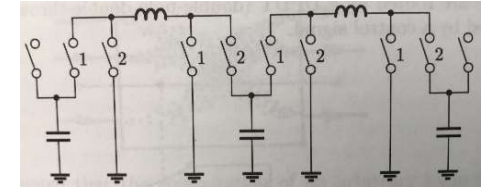
- Not yet fully-reversible technique; limited efficiency
- Combinational only; conjectured reversible *sequential* logic impossible

Hall, 1992 and Merkle, 1992 (DOIs:10.1109/PHYCMP.1992.615549;
10.1109/PHYCMP.1992.615546)

- General-purpose reversible methods, but for combinational logic only

Younis & Knight (MIT), 1993 (<http://dl.acm.org/citation.cfm?id=163468>)

- First fully-reversible, fully-adiabatic *sequential* circuit technique (CRL)



Adiabatic Circuits in CMOS: History, cont.

Younis, Saed G., and T. F. Knight. "Asymptotically zero energy split-level charge recovery logic." In *Low-Power CMOS Design*, Chandrakasan, A. and R. Brodersen, eds., IEEE Press (1994): pp. 253-258. On IEEExplore at DOI: [10.1109/9780470545058.sect8](https://doi.org/10.1109/9780470545058.sect8)



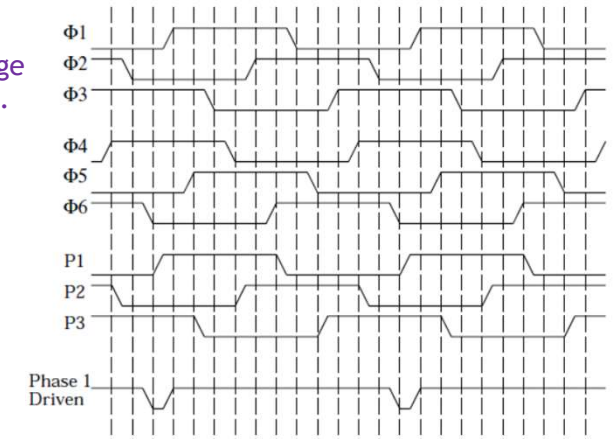
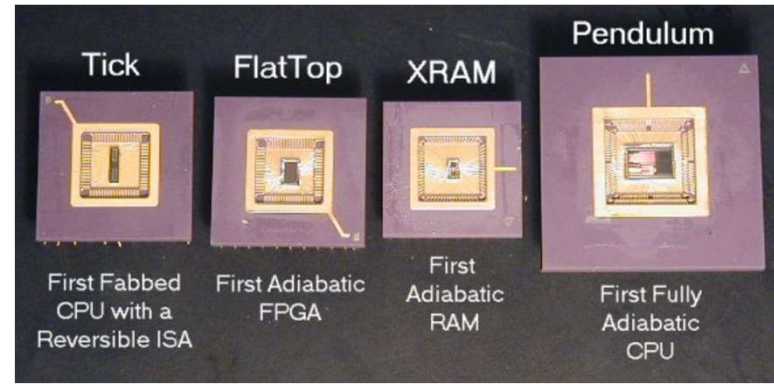
Younis & Knight, 1994:

- Simplified 3-level adiabatic CMOS design family (SCRL) →
 - However, the original version of SCRL contained a small non-adiabaticity bug which I discovered in 1997.
 - This problem is easily fixed, however.

Saed G. Younis, "Asymptotically zero energy computing using split-level charge recovery logic," Ph.D. thesis, MIT, 1994. <http://hdl.handle.net/1721.1/11620>

Subsequent work at MIT, 1995-99:

- By myself and fellow students.
- Various chips designed using SCRL →
- Reversible processor architectures.



Substantial literature throughout the late 90s / early 2000s...

- Too many different papers / groups to list them all here!
 - Most of the proposed schemes are not truly/fully adiabatic, though...

Researchers recently active in adiabatic circuits include:

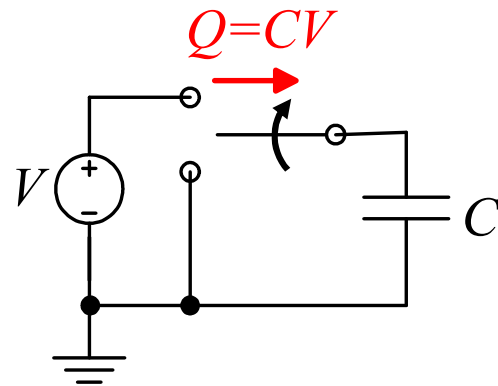
- A couple I know in the US:
 - Greg Snider (Notre Dame)
 - Himanshu Thapliyal (U. Kentucky)
- Also some groups in Europe, India, China, Japan...
- My group at Sandia (new work reported on a later slide)

Conventional vs. Adiabatic Charging

For charging a capacitive load C through a voltage swing V

Conventional charging:

- Constant *voltage* source

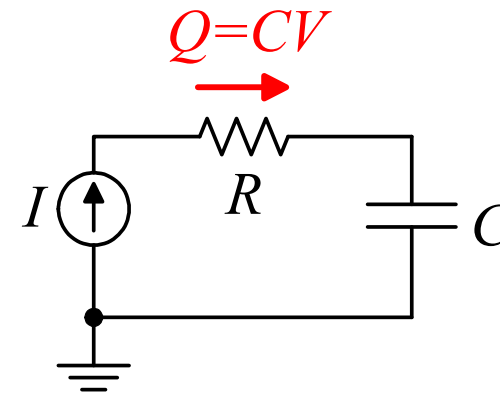


- Energy dissipated:

$$E_{\text{diss}}^{\text{conv}} = \frac{1}{2} CV^2$$

Ideal *adiabatic* charging:

- Constant *current* source



- Energy dissipated:

$$E_{\text{diss}}^{\text{adia}} = I^2 R t = \frac{Q^2 R}{t} = CV^2 \frac{RC}{t}$$

Note: Adiabatic charging beats the energy efficiency of conventional by advantage factor:

$$A = \frac{E_{\text{diss}}^{\text{conv}}}{E_{\text{diss}}^{\text{adia}}} = \frac{1}{2} \frac{t}{RC}$$

Adiabatic Charging via MOSFETs

A simple voltage ramp can *approximate* an ideal constant-current source.

- Note that the load gets charged up *conditionally*, if the MOSFET is turned on (gate voltage $V_g \gtrsim V + V_t$) throughout the ramp.
- V_t is the transistor's threshold, typically $< 1/2$ volt

Can discharge the load later using a similar ramp.

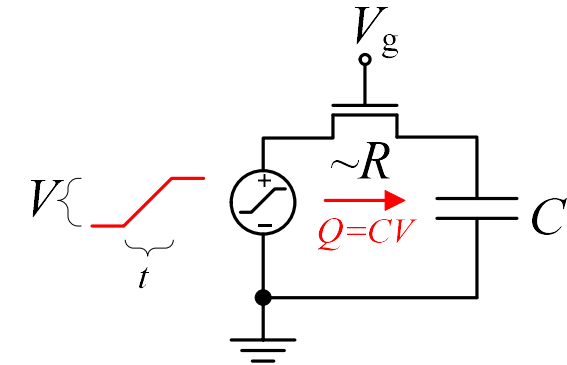
- Either through the same path, or a different path.

$$t \gg RC \Rightarrow E_{\text{diss}} \rightarrow CV^2 \frac{RC}{t}$$

$$t \ll RC \Rightarrow E_{\text{diss}} \rightarrow \frac{1}{2} CV^2$$

The (ideal) operation of this circuit approaches *physical reversibility* ($E_{\text{diss}} \rightarrow 0$) in the limit $t \rightarrow \infty$, but *only* if a certain *precondition* on the initial state is met (namely, $V_g \gtrsim V_{\text{max}} + V_t$)

- How does the possible physical reversibility of this circuit relate to its *computational* function, and to some *appropriate* concept of logical reversibility?
- Traditional (Landauer/Fredkin/Toffoli) reversible computing theory does not adequately address this question, so, we need a more powerful theory!
 - The theory of **Generalized Reversible Computing** (GRC) meets this need.



Exact formula for linear ramps:

$$E_{\text{diss}} = s[1 + s(e^{-1/s} - 1)]CV^2$$

given *speed fraction* $s = RC/t$.

See [arxiv:1806.10183](https://arxiv.org/abs/1806.10183) for the full GRC model.

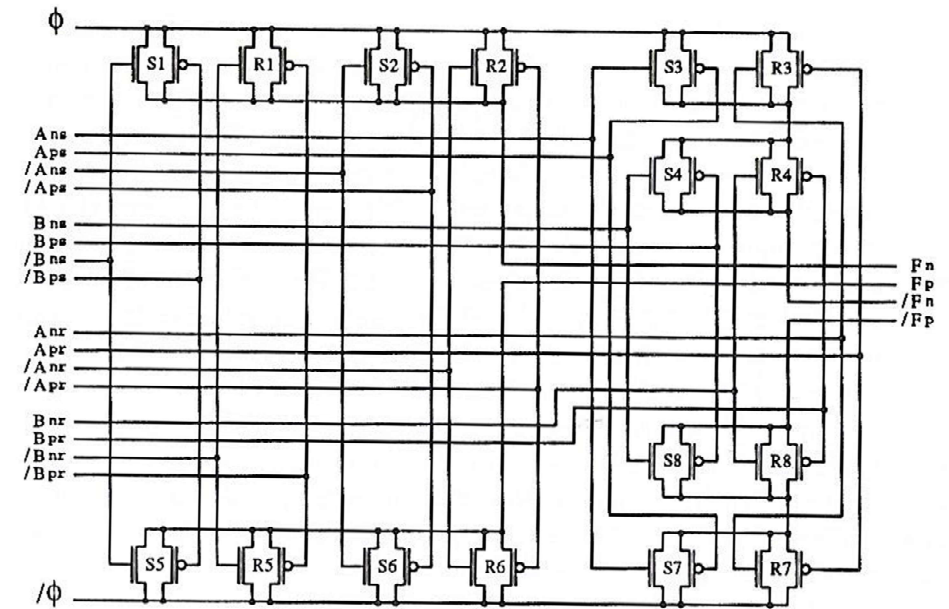
Early Examples of Fully Adiabatic CMOS Logic Families

S. G. Younis and T. F. Knight, Jr., “Practical implementation of charge recovering asymptotically zero power CMOS,” in Research on Integrated Systems: Proc. 1993 Symp., C. Ebeling and G. Borriello, Eds. Cambridge: MIT Press, Feb. 1993, pp. 234–250.

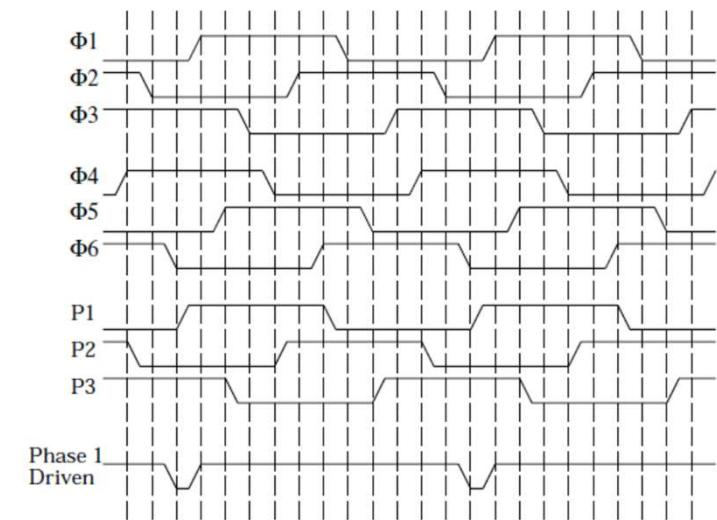
- First fully adiabatic, general sequential CMOS logic family.
- Four clock phases, four transitions per clock cycle.
- Quad-rail logic encoding.
- Slightly generalized by the 2LAL logic family (Frank, 2000).
- Dynamic logic.

S. G. Younis, “Asymptotically Zero Energy Computing Using Split-Level Charge Recovery Logic,” Ph.D. thesis, Massachusetts Institute of Technology, June 1994. dspace.mit.edu/handle/1721.1/11620

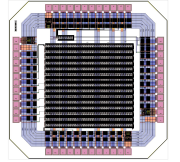
- Simplified hardware designs compared to CRL.
 - Single-rail logic is possible.
- Several clocking variants, including “static” versions.
- Contains a minor non-adiabatic/non-static bug, I discovered in ‘97.
 - Easily fixed, however, by adding 1 extra transistor per logic gate.



Younis & Knight ‘93: CRL 2-input NAND gate.



Younis ‘94: Clocks for 24-tick “static” SCRL



Perfectly Adiabatic Reversible Computing in CMOS

To approach ideal reversible computing in CMOS...

We must aggressively eliminate *all* sources of non-adiabatic dissipation, including:

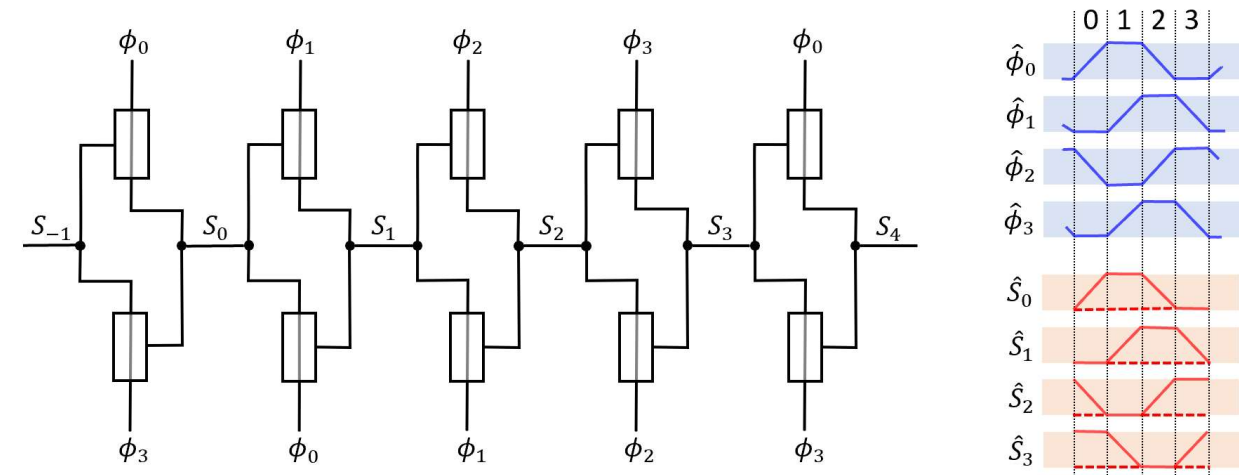
- Diodes in charging path, “sparking,” “squelching,”
 - Eliminated by “**truly, fully adiabatic**” design. (E.g., CRL, 2LAL).
 - Can suffice to get down to a few aJ (10s of eV) even *before* voltage optimization.
- Voltage level mismatches that dynamically arise on floating nodes before reconnection.
 - Eliminated by static, “**perfectly adiabatic**” design. (E.g., S2LAL).

We must also aggressively minimize standby power dissipation from leakage, including:

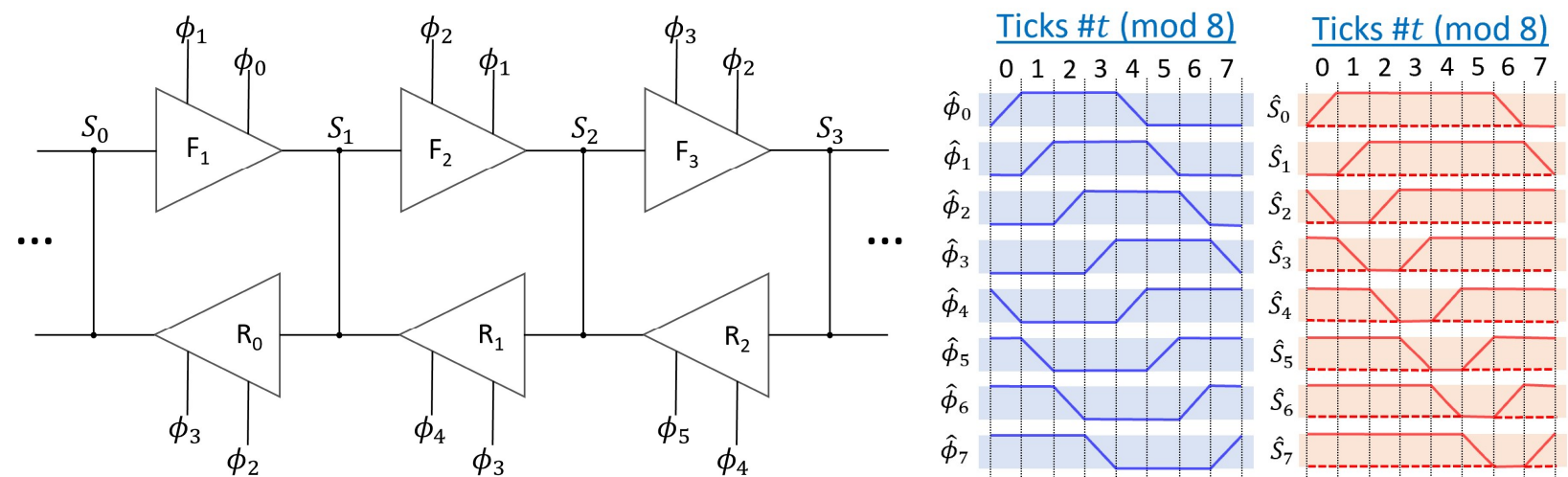
- Subthreshold channel currents.
 - Ultra-low- T (e.g. 4K) operation helps with this.
- Tunneling through gate oxide.
 - E.g., use thicker gate oxides.

Note: (Conditional) logical reversibility *follows from* perfect adiabaticity.

Shift Register Structure and Timing in 2LAL



Shift Register Structure and Timing in S2LAL



(arxiv:2009.00448)

An SRC-funded study done at the University of Florida (2004)



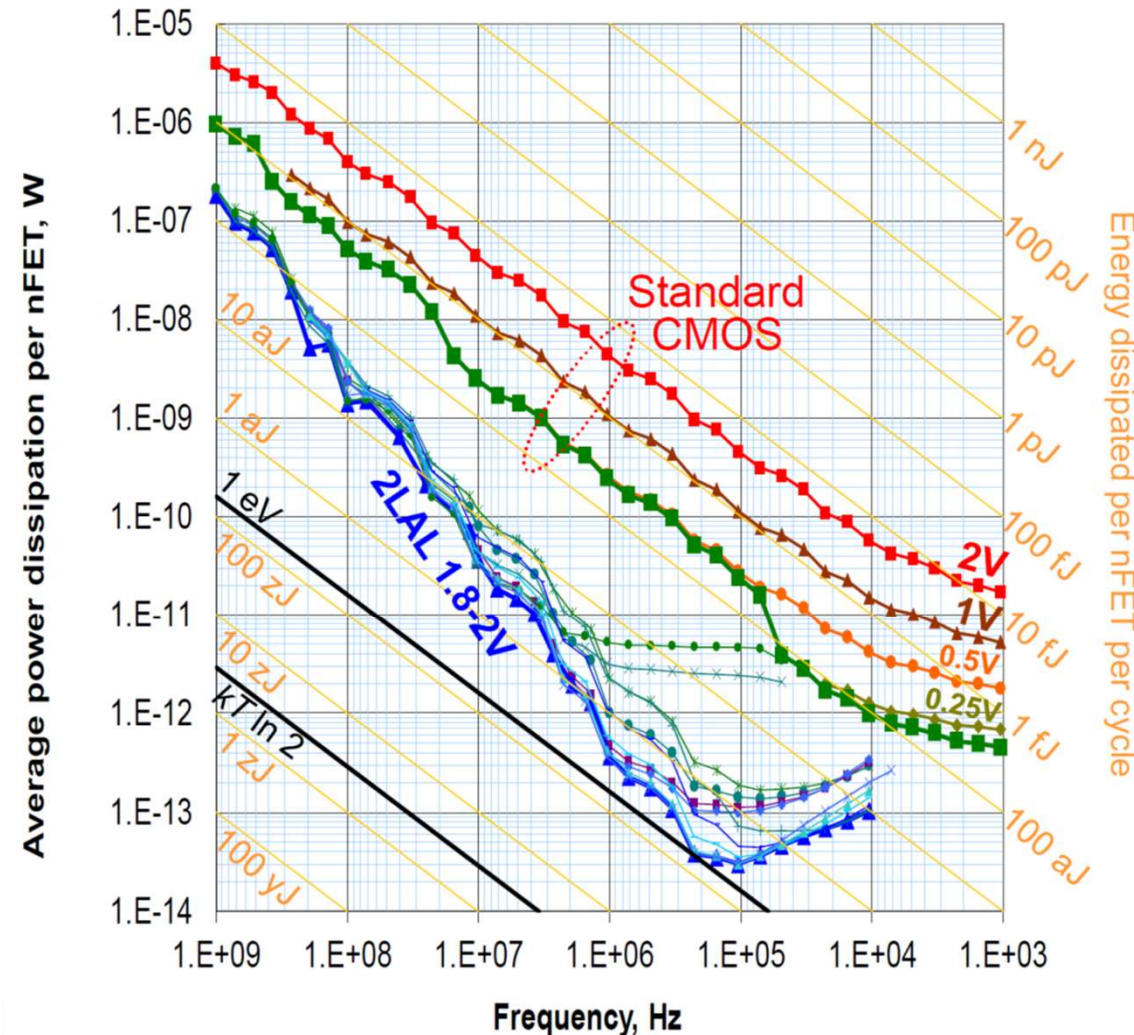
Semiconductor
Research
Corporation



Simulation Results (Cadence/Spectre) **UF** UNIVERSITY of FLORIDA

Power vs. freq., TSMC 0.18, Std. CMOS vs. 2LAL

2LAL = Two-level adiabatic logic (invented at UF, '00)



- Graph shows per-FET power dissipation vs. frequency
 - in an 8-stage shift register.
- At moderate freqs. (1 MHz),
 - Reversible uses $< 1/100^{\text{th}}$ the power of irreversible!
- At ultra-low power levels (1 pW/transistor)
 - Reversible is $100 \times$ faster than irreversible!
- Minimum energy dissipation per nFET is **< 1 electron volt!**
 - $500 \times$ lower dissipation than best irreversible CMOS!
 - $500 \times$ higher computational energy efficiency!
- Energy transferred per nFET per cycle is still on the order of 1-10 fJ (10-100 keV)
 - So, energy recovery efficiency is at least 99.99%!
 - Quality factor $Q > 10,000!$
 - Note this does not include any of the parasitic losses associated with power supply and clock distribution yet, though

Simulation results *appeared* to show that 2LAL in TSMC 180nm could get to as low as 1 ev (!) dissipation/FET/ clock cycle.

We now believe (thanks to a current NSCI-funded study at Sandia) that that specific result was most likely unrealistic, because the BSIM3 models we had in '04 (we think) probably substantively underestimated the actual gate leakage resulting from tunneling.

- We think that specific BSIM3 model did not capture gate leakage at all.

However, we do still believe that, in a real process that was well optimized for low leakage, we would be able to achieve similarly impressive results to this.

Latest Results from the “Adiabatic Circuits Feasibility Study” Simulation Efforts at Sandia, funded via NSCI (2017-2021)



Created schematic-level fully-adiabatic designs for Sandia’s in-house processes, including:

- Older, 350 nm process (**blue** curve)
 - FET widths = 800 nm
- Newer, 180 nm process (**orange, green** curves)
 - FET widths = 480 nm

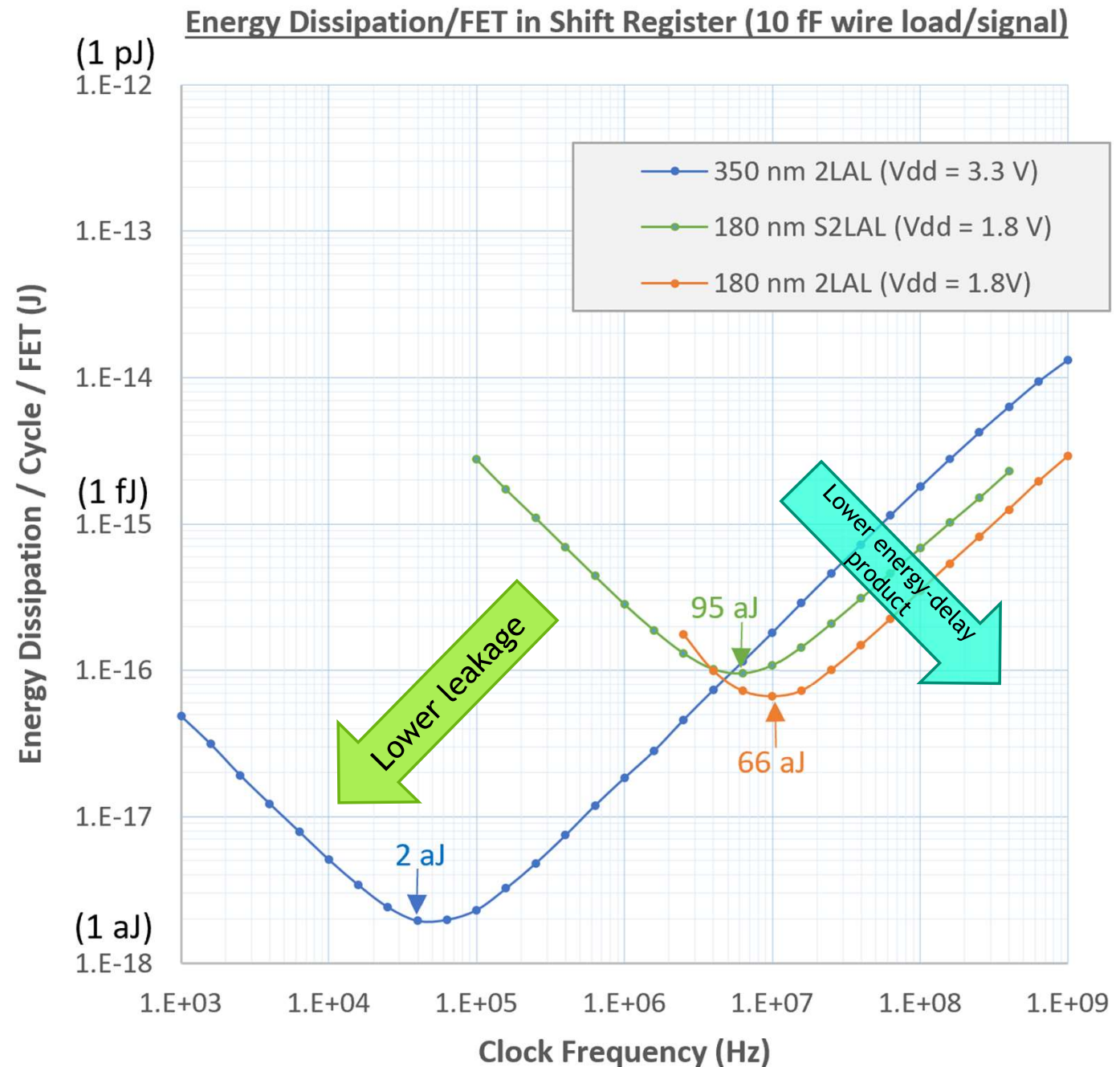
Plotted energy dissipation per-transistor in shift registers at 50% activity factor (alternating 0/1)

- 2LAL (**blue, orange** curves)
- S2LAL (**green** curve)

In all of these Cadence/Spectre simulations,

- We assumed a 10 fF parasitic wiring load capacitance on each interconnect node.
- Logic supply (V_{dd}) voltages were taken at the processes’ nominal values.
 - 3.3V for the 350nm process; 1.8V in the 180nm process.

We expect these results could be significantly improved by exploring the parameter space over possible values of V_{dd} and V_{sb} (substrate bias).



Minimum Energy Scaling for Adiabatic CMOS

From M. Frank & K. Shukla,
doi:10.3390/e23060701



Appendix A. Minimum-Energy Scaling for Classical Adiabatic Technologies

In this appendix, we briefly present the derivation for the scaling of minimum energy dissipation for reversible technologies such as RA-CMOS (Section 2.3.1) that obey classic adiabatic scaling and that can be characterized in terms of relaxation and equilibration timescales.⁴⁴

First, we assume (as is the case for “perfectly adiabatic” technologies such as [48]) that the total energy dissipation per clock cycle E_{diss} in a reversible circuit can be expressed as a sum of *switching losses* and *leakage losses*,

$$E_{\text{diss}} = E_{\text{sw}} + E_{\text{lk}}, \quad (\text{A1})$$

and further, that switching and leakage losses depend on the signal energy E_{sig} and transition time t_{tr} approximately as follows:

$$E_{\text{sw}} \simeq E_{\text{sig}} \cdot c_{\text{sw}} \cdot \frac{\tau_r}{t_{\text{tr}}}, \quad (\text{A2})$$

$$E_{\text{lk}} \simeq E_{\text{sig}} \cdot c_{\text{lk}} \cdot \frac{t_{\text{tr}}}{\tau_e}, \quad (\text{A3})$$

where τ_r, τ_e are the relaxation and equilibration timescales, respectively, and $c_{\text{sw}}, c_{\text{lk}}$ are small dimensionless constants characteristic of a particular reversible circuit in a specific family of technologies, such as [48]. In practice, although these specific formulas are only approximate, they approach exactness in the regime $\tau_r \ll t_{\text{tr}} \ll \tau_e$.

Then, now treating (A2), (A3) as exact, we can write:

$$E_{\text{diss}} = E_{\text{sig}} \left(c_{\text{sw}} \tau_r \cdot \frac{1}{t_{\text{tr}}} + \frac{c_{\text{lk}}}{\tau_e} \cdot t_{\text{tr}} \right). \quad (\text{A4})$$

We can collect the constants, absorbing them into adjusted timescales $\tau'_r = c_{\text{sw}} \tau_r$ and $\tau'_e = \tau_e / c_{\text{lk}}$, so

$$E_{\text{diss}} = E_{\text{sig}} \left(\tau'_r \cdot \frac{1}{t_{\text{tr}}} + \frac{1}{\tau'_e} \cdot t_{\text{tr}} \right). \quad (\text{A5})$$

Setting the derivative of (A5) with respect to t_{tr} equal to zero, we find that E_{diss} is minimized when

$$\tau'_r \frac{1}{t_{\text{tr}}^2} = \frac{1}{\tau'_e}, \quad (\text{A6})$$

Upshot for CMOS: As each device's leakage conductance I_{off} is decreased, the equilibration timescale τ_e increases, and the technology's minimum energy (given perfectly adiabatic, reversible designs) scales down with square-root proportionality.

$$E_{\text{diss,min}} \propto \frac{1}{\sqrt{\tau_e}} \propto \sqrt{I_{\text{off}}}$$

or in other words, when

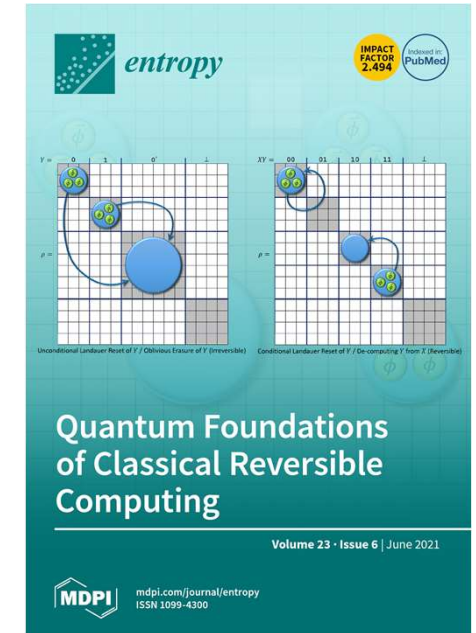
$$t_{\text{tr}} = \sqrt{\tau'_r \tau'_e}, \quad (\text{A7})$$

at which point E_{sw} and E_{lk} are equal. The minimum energy dissipation per cycle is then

$$E_{\text{diss}} = 2E_{\text{sig}} \sqrt{\frac{\tau'_r}{\tau'_e}}. \quad (\text{A8})$$

Thus, for any given reversible circuit design in a family of technologies with given values of the constants $c_{\text{sw}}, c_{\text{lk}}$, in order for E_{diss} to approach 0 as the technology develops, we must have that the ratio of equilibration/relaxation timescales $\tau_e / \tau_r \rightarrow \infty$, and, if the relaxation timescale τ_r is fixed, this implies that also the (minimum-energy) value of the transition time $t_{\text{tr}} \rightarrow \infty$. These requirements were mentioned in Section 2.3.1.

More specifically, in order to increase the peak energy efficiency of a reversible circuit by a factor of $N \times$, in a given family of technologies obeying classic adiabatic scaling, this requires that the timescale ratio τ_e / τ_r must be increased by $N^2 \times$, and (assuming τ_r is fixed) the transition time t_{tr} for minimum energy will increase by $N \times$.



Basic Requirements for Fully Adiabatic Operation

No diodes in charging paths!

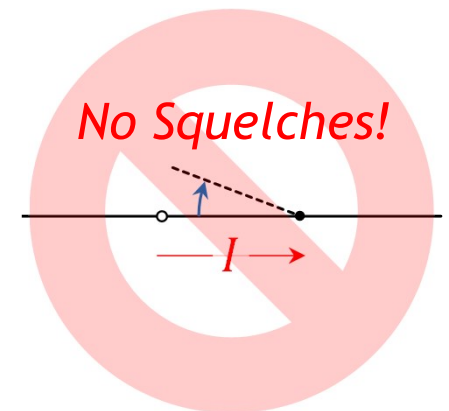
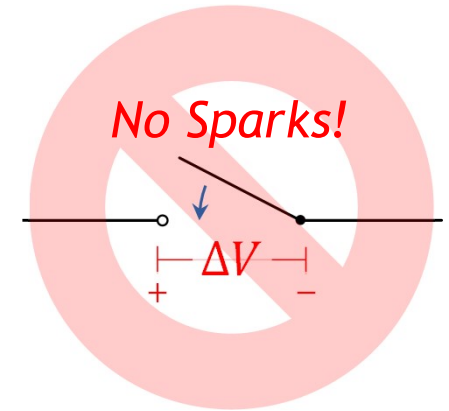
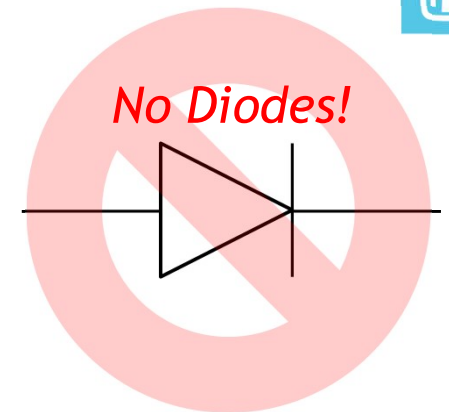
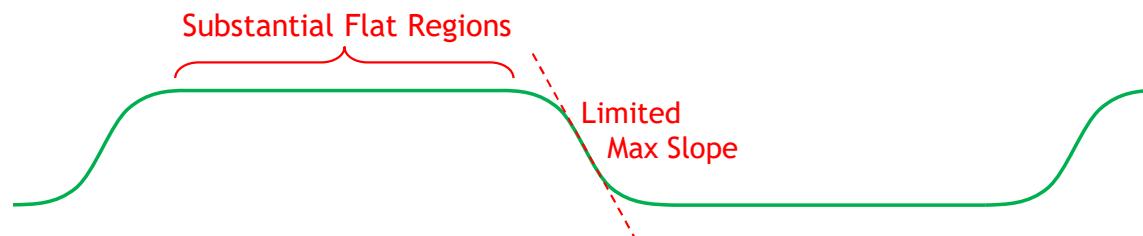
- All diodes have a built-in voltage drop for fundamental thermodynamic reasons.

Operate all switches (*e.g.*, FETs) with a “dry-switching” discipline:

- Never turn on (close) a switch when there is a significant voltage difference $\Delta V \neq 0$ between its terminals.
 - Leads to a sudden, non-adiabatic flow of current (a.k.a. “sparking”).
 - More generally: No rapid voltage changes.
- Never turn off (open) a switch when there is a significant current flow $I \neq 0$ through the switch.
 - Leads to non-adiabatic losses as switch is (non-instantaneously) turning off (a.k.a. “squelching”).
 - Resistance through switch increases during turnoff \rightarrow voltage drop increases \rightarrow non-adiabatic loss across voltage drop.
 - Exception: If path is low inductance and there is an alternate path for the current.

Use quasi-trapezoidal driving waveforms (no steep edges; flat tops and bottoms).

- This is necessary to obey the other rules.



Why Static Adiabatic Logic?

In non-static (*i.e.*, *dynamic*) logic styles, by definition, some circuit nodes are allowed to *float* dynamically (*i.e.*, without any direct tie to source) for at least part of the time.

- *E.g.*, this happens in a dynamic random-access memory (DRAM) cell.

The problem with having floating nodes is that their voltages may *vary from their ideal level* while they are isolated, for example, due to:

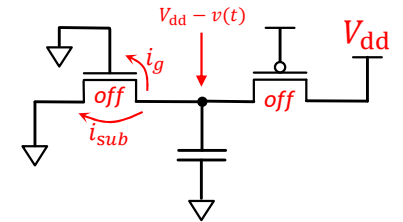
- *Voltage drift* due to leakage currents to sources at different levels through nominally turned-off devices. Includes:
 - Subthreshold leakage current $i_{sub}(t)$ across the channel of a device below threshold.
 - Gate leakage current $i_g(t)$ due to tunneling through the gate oxide.
- *Voltage sag* due to capacitive voltage-division effects involving parasitic capacitive couplings to nearby nodes with time-varying voltages.

If a floating node with capacitance C has a voltage disparity of ΔV from a given reference level at the time that it is reconnected to a source at that level,

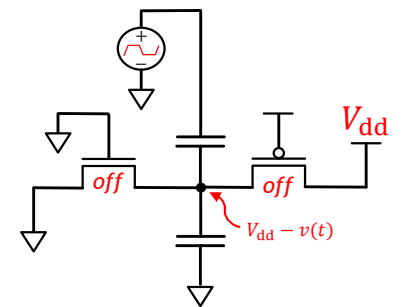
- Then there will be a sudden non-adiabatic “sparking” event dissipating $C(\Delta V)^2/2$ energy at the time of reconnection.

Avoiding these sparking events would require very precise engineering of all the possible paths for leakage and sag (*e.g.* to ensure the effects cancel)...

- OR, we could just design a fully static logic family! **← Much easier!**



Voltage drift due to leakage



Voltage sag due to capacitive coupling to nearby varying nodes

Rules for Fully Static Operation

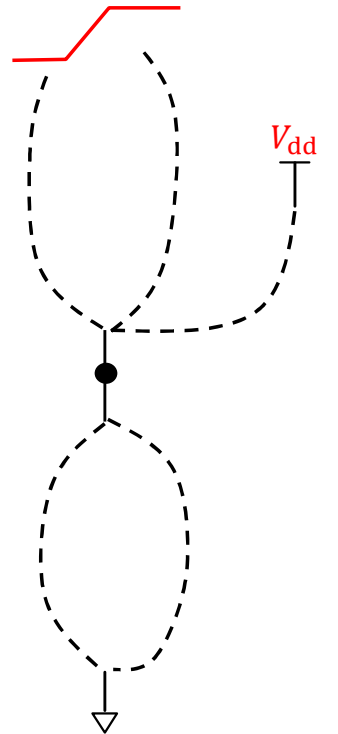


At *all times*, *each* internal node of the circuit must be connected to a voltage reference in one of the following manners:

1. Connected via a (medium-impedance) path through turned-on transistor(s) to a single constant-voltage reference;
2. Connected via a (medium-impedance) path through turned-on transistor(s) to a single variable-voltage reference;
3. Connected in a way that is actively transitioning (in either direction) between conditions 1 & 2 above,
 - with one path in the process of being connected while the other is in the process of being disconnected, and
 - where, at any given time throughout the transition, at least one path has no more than medium impedance, and
 - where, throughout the transition period, the level of the variable-voltage reference in question is being held constant at the same level as the constant-voltage rail;
4. Connected in a way that is (similarly) actively transitioning between two different paths to a single supply reference (whether it is constant-voltage or variable-voltage).

Where “medium impedance” here means below some reasonable upper limit (*e.g.* 100 k Ω).

- And, all paths that are nominally “off” should have a much higher impedance, *e.g.*, $\gg 1$ M Ω .
- The higher the off-state resistance, the lower will be the minimum energy.



Notations and Conventions Used (slide 1 of 2)



Two nominal voltage levels: 0 V (GND, “low”) and $V_{\text{dd}} \gtrsim 2|V_{\text{t}}|$ (“high”).

Divide time into equal, discrete intervals called *ticks*, each of duration $\bar{\tau}_{\text{tr}}$, and numbered consecutively.

- Every *transition* between nominal levels is required to fit entirely within a tick,
 - so, the actual transition time τ_{tr} is upper-bounded by the tick length, $\tau_{\text{tr}} \leq \bar{\tau}_{\text{tr}}$.

The active energy dissipation from any given adiabatic transition is as follows:

$$E_{\text{a}} = \xi_{\text{tr}} C_{\text{L}} V_{\text{dd}}^2 \frac{RC_{\text{L}}}{\tau_{\text{tr}}},$$

where:

- ξ_{tr} is a *shape factor* that accounts for the departure of the ramp shape from the ideal;
- C_{L} is the capacitive load of the node that is transitioning;
- R is the effective resistance of the charging path.

The clock period τ_{p} is an integer number n of ticks, $\tau_{\text{p}} = n\bar{\tau}_{\text{tr}}$.

- Thus, the clock frequency is

$$f = (n\bar{\tau}_{\text{tr}})^{-1}.$$

- Ticks within a cycle are numbered modulo n (i.e., $0, \dots, n-1$).

Notations and Conventions Used (slide 2 of 2)

In the logic styles we'll discuss, any given logic *symbol* L (e.g., 0 or 1) is represented by a complementary *signal pair*.

- Thus, for k -valued logic we require $2k$ signals.
- Normally we have just $k = 2$ symbols, $L \in \{0,1\}$.

Possible conditions for a given signal pair (when valid) are *active* or *inactive*.

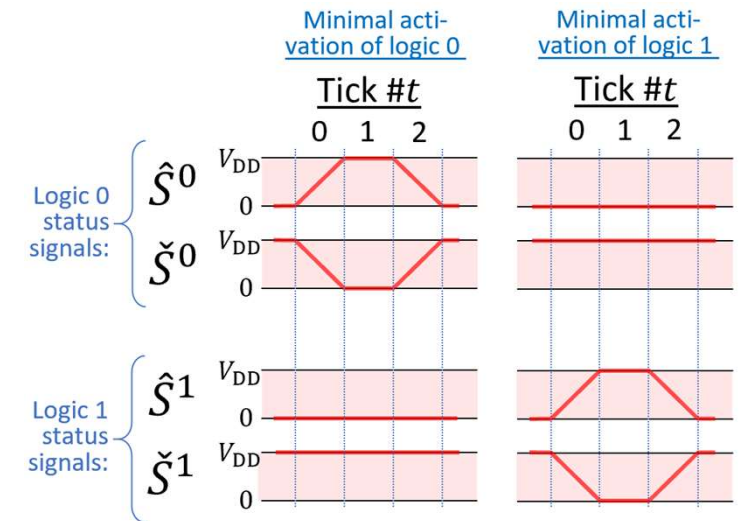
- One of the signals in each pair is *active-high*; the other is *active-low*.
 - When in the active state, we say the pair is *actively representing* the corresponding logic symbol L .
- The signal pair may feed the control terminals of a CMOS transmission gate.
 - The active-high signal controls the nFET, and the active-low signal controls the pFET.
 - Thus, the transmission gate is turned ON (conducting) when the signal pair is active.
 - The body terminals of the FETs should be separately biased (not tied to either channel terminal).
 - Can be used to increase device thresholds if desired.

The following notation is used for a signal pair:

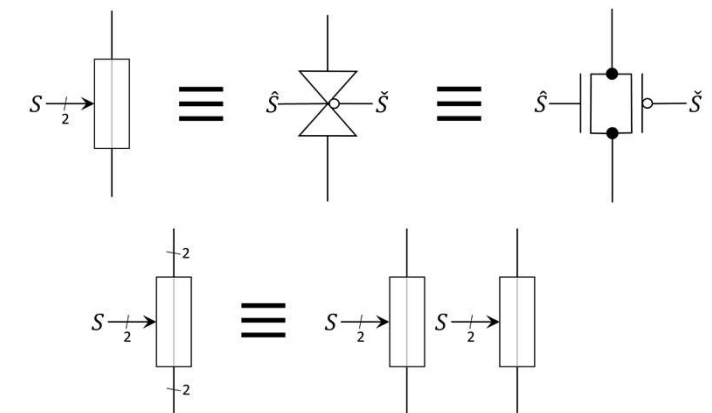
$$S_{t_b, t_e}^L = (\hat{S}_{t_b, t_e}^L, \check{S}_{t_b, t_e}^L)$$

where:

- $\hat{}, \check{}$ accents denote active-high and active-low signals, respectively.
 - No accent denotes the pair.
- L (if present) denotes the logic symbol the signal pair is representing.
- t_b, t_e (if present) denote the transitional (*begin* and *end*) ticks of the active period.



Examples of minimal activations

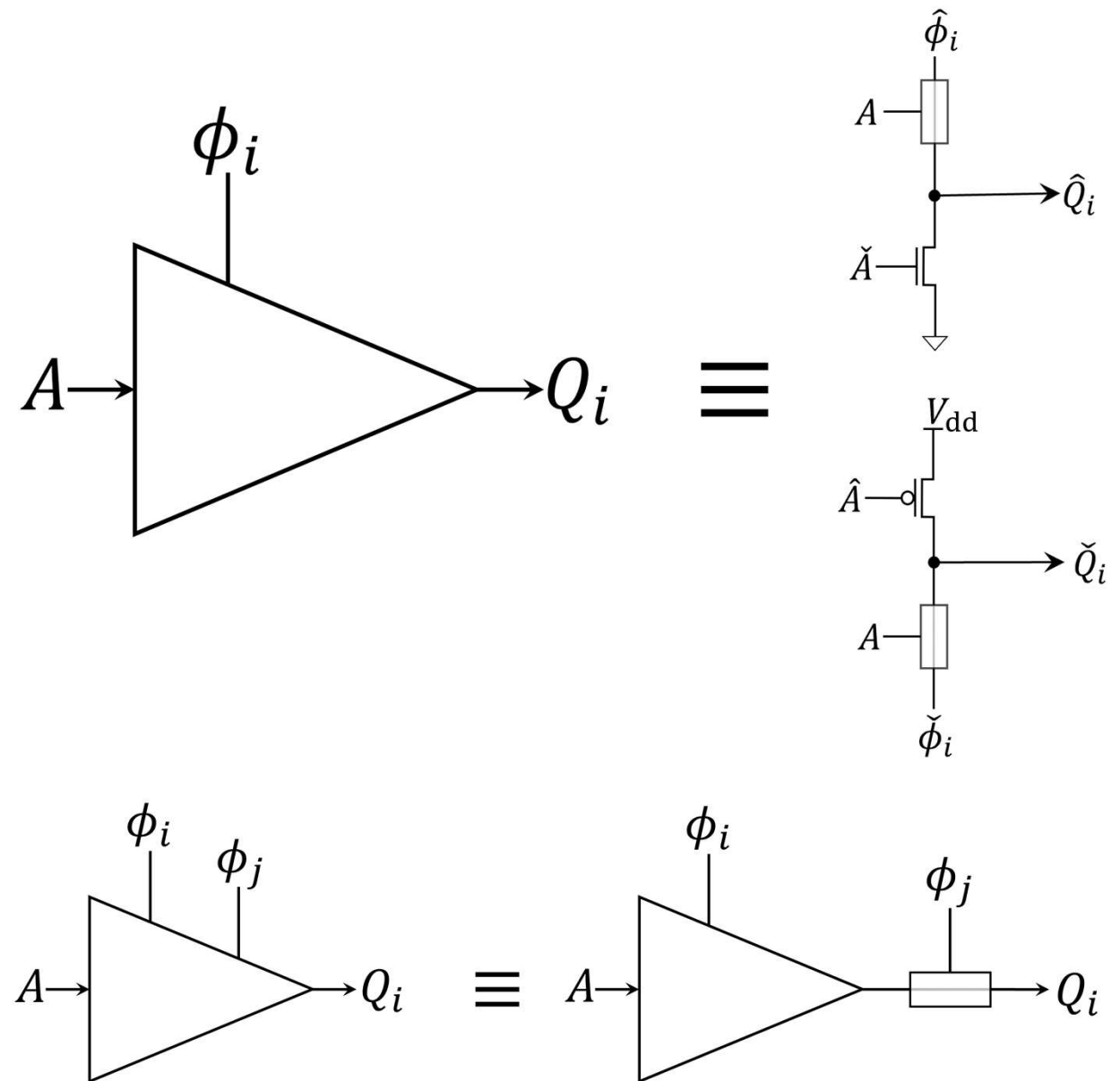


Transmission gate symbols

Basic Elements of S2LAL

Unlatched & Latching Static Adiabatic Buffers

- Unlatched version exchanges control of output between clock and fixed supply, depending on activity of input.
 - Handoff should only happen when levels match.
 - Athas '94 called this same element an *adiabatic amplifier*.
 - Athas, W.C., *et al.* "Low-power digital systems based on adiabatic-switching principles," *IEEE Trans. VLSI Sys.* 2(4):398–407, 1994. [doi:10.1109/92.335009](https://doi.org/10.1109/92.335009)
- *Latching* version uses an out-of-phase clock to latch (or unlatch!) the output.
 - NOTE: This requires additional higher-level structure to make the resulting circuit fully static!



S2LAL Reversible Pipeline Structure

Paired forward and reverse stages:

- Forward stages activate to compute *later* signals from *earlier* ones.
- Reverse stages *de-activate* to *de-compute earlier* signals from *later* ones.

Every signal S_i must stay active for (at least) 5 ticks:

- Provides sufficient time for the following sequence of steps:
 - (1) Activate forwards stage F_{i+1} , (2) Activate reverse stage R_i , (3) Handoff control of S_i from F_i to R_i , (4) Deactivate forwards stage F_i , (5) Deactivate reverse stage R_{i-1} .

Add 3 ticks for transitions & inactive handoff:

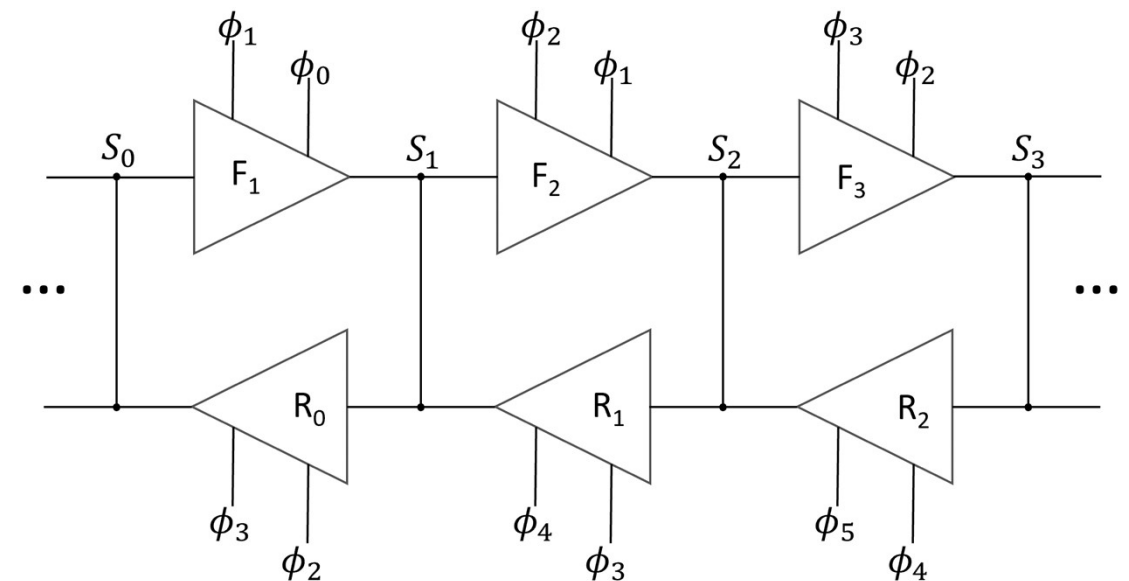
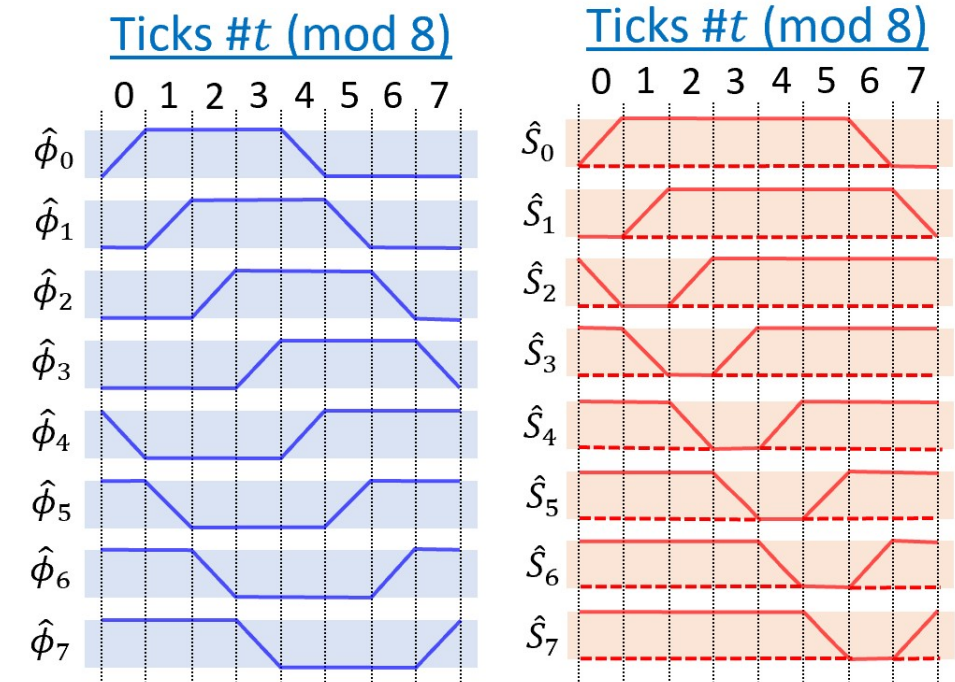
- Total cycle length = **8 ticks** minimum.

Note control of each signal S_i is handed off to forward stage F_i on ticks $\#i - 1$, and to reverse stage R_i on ticks $\#i + 3$.

- Signal S_i goes valid on ticks $\#i$ and invalid (inactive) on ticks $\#i + 6$.

For general logic, functions must be invertible.

- Optimizing whole pipeline gets into reversible algorithm design: Considered out of scope for this particular paper.



S2LAL Logic Gates

14-transistor AND gate, 16-transistor OR gate.

- Carefully designed to ensure that each internal node is always connected to either constant or variable source.
- The structures shown are minimal, given the design constraints.

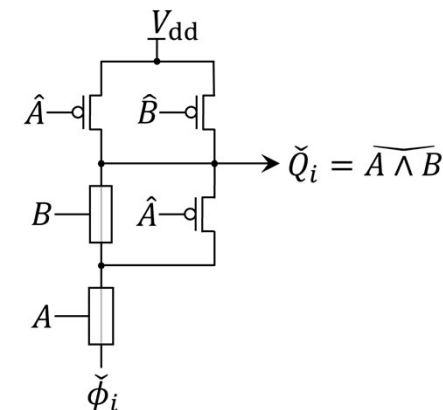
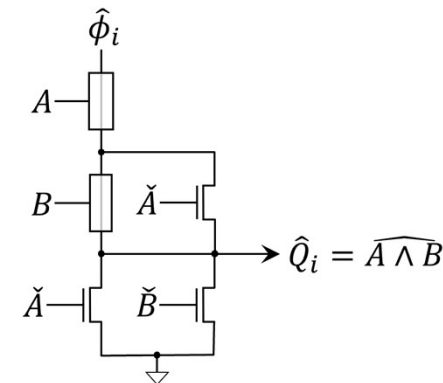
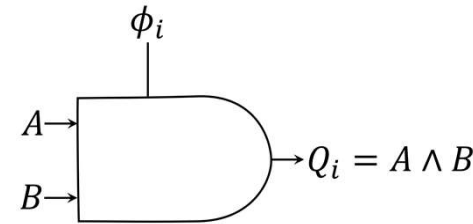
Inverting gates are done easily, by using signal pairs for complementary symbols:

- $\text{NOT}(A^1) = \text{BUFFER}(A^0)$
- $\text{NAND}(A^1, B^1) = \text{OR}(A^0, B^0)$
- $\text{NOR}(A^1, B^1) = \text{AND}(A^0, B^0)$

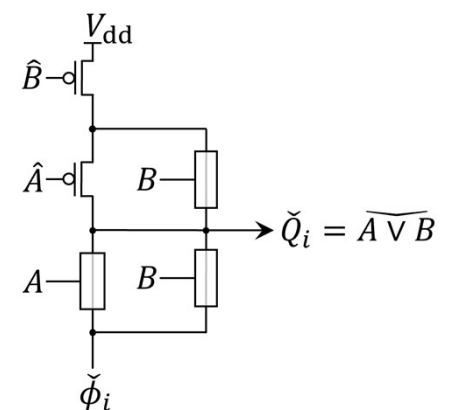
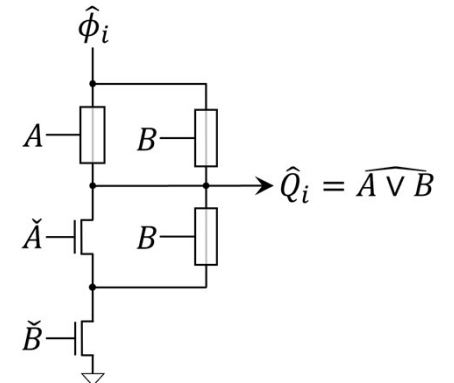
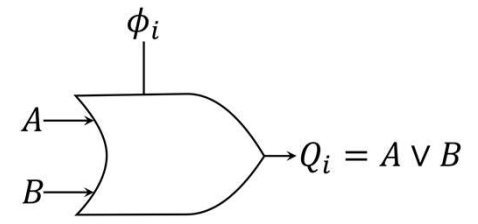
Also! Erik DeBenedictis invented an optimization to S2LAL that can compute the inverses as-needed, rather than keeping both the 0,1 signal pairs around:

- See <https://zettaflops.org/zf004/>.

AND



OR



Resonator design effort, in progress...

See Frank *et al.* “Exploring the Ultimate Limits of Adiabatic CMOS”, 38th IEEE Int’l Conf. on Computer Design (ICCD’20), [10.1109/ICCD50377.2020.00018](https://doi.org/10.1109/ICCD50377.2020.00018)



Goal of this effort:

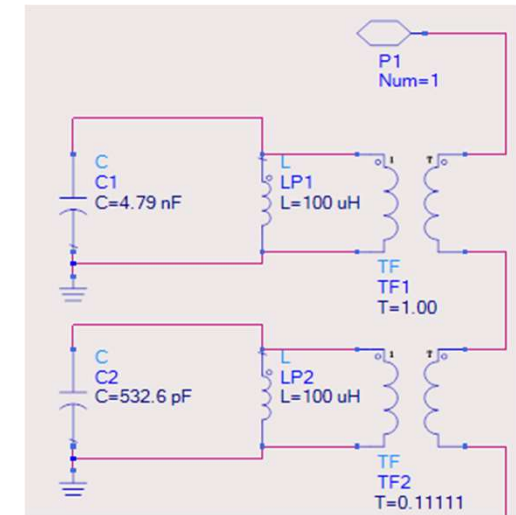
- Design & validate a high-efficiency resonant oscillator (for low-to-medium RF frequencies) that approximates a trapezoidal output voltage waveform.

Innovative design concept:

- Transformer-coupled** assemblage of LC tank circuits with resonant frequencies corresponding to odd multiples of the fundamental frequency, excited in the right relative amplitudes to approximate the target wave shape

Some detailed requirement specifications:

- Initial target operating point: 230 kHz, 1.8V (optimal point for minimum dissipation in the UF study) **(Has been met.)**
 - However, our circuit technique should be adaptable over a wide range of frequencies and voltages.
- Tops and bottoms of trapezoidal wave should be within $\leq 5\%$ of flatness throughout $\frac{1}{4}$ clock period. **(Met.)**
- The 10-90% rise/fall time should be between 75 & 100% of its nominal value (80% of $\frac{1}{4}$ clock period) **(Met.)**
- Efficiency goals:
 - Quality factor of resonator during unpowered ring-down should be $\geq 1,000$. **(Met. Simulated value: $\sim 3,000$.)**
 - Total energy dissipation per cycle during steady-state powered operation should be $\leq 1\%$ of magnetically-stored energy in the resonator, when the oscillator is running in isolation. (Still needs validation.)
 - Total energy dissipation per cycle during steady-state powered operation should be $\leq 10\%$ of the capacitively-stored energy on an appropriately-sized model (RC) load, when the oscillator is coupled to the load. (Needs validation.)

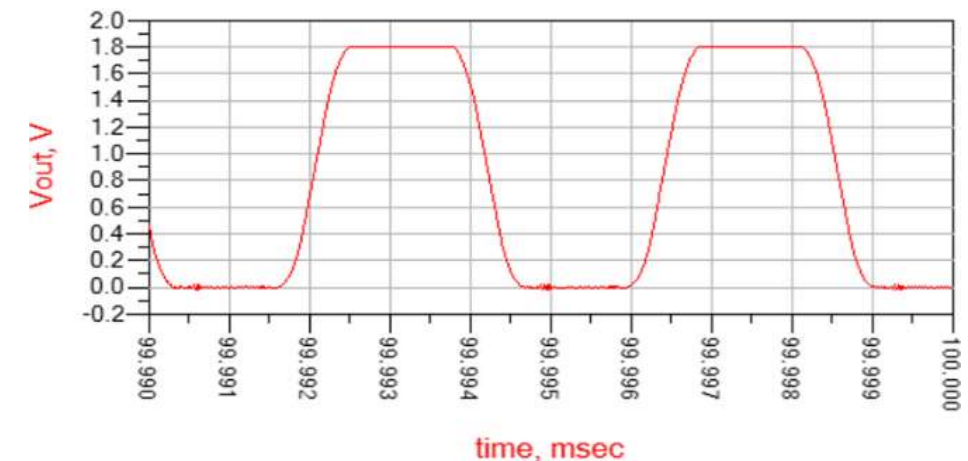


A number of significant design challenges that have been encountered so far:

- How to tune the relative amplitudes of the component resonant modes **(Solved.)**
- How to prevent phase drift and transfer of energy between modes **(Solved.)**
- Identifying/tailoring components to have precise-enough L , C values
- Designing a driver circuit that meets efficiency goals during steady-state operation
- Packaging & integration for a complete system including a resonator & a 2LAL die.

A patent application has been filed on our resonator design.

- We invite industry firms to partner with us under NDA/CRADA.



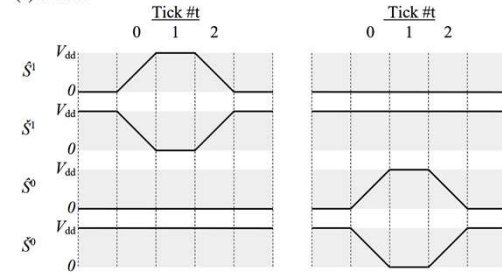
Q2LAL (by Erik DeBenedictis)

<https://ar.zettaflops.org/CATC/Q2LAL.pdf>



Cuts complexity of S2LAL roughly in half!

(a) S2LAL



(b) Q2LAL

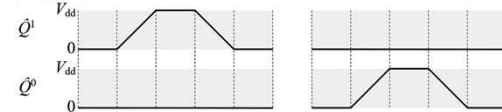
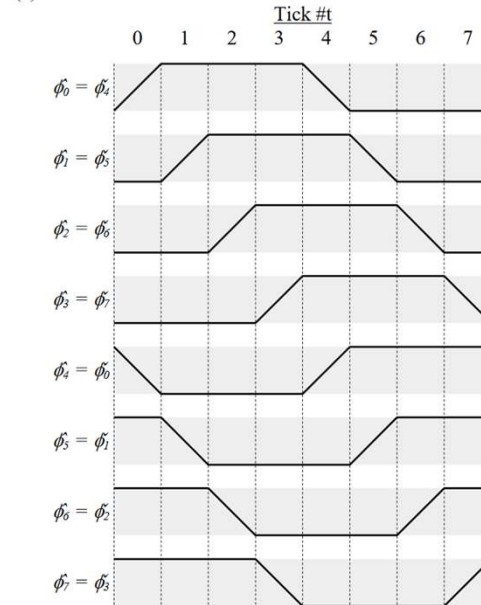
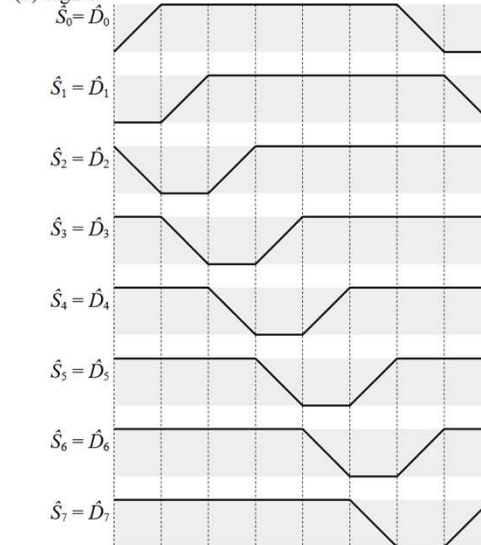


Fig. 1. Signal waveforms. (a) Predecessor S2LAL is dual or quad rail. (b) Q2LAL is dual rail. Notation from [4].

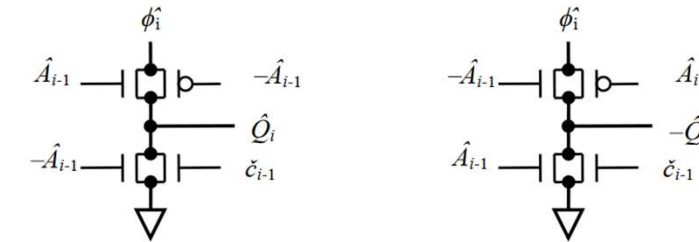
(a) Clocks



(b) Signals



(c) Q2LAL: replace cups; add extra clamp transistor



(d) helper signal for clamp; does not depend on data

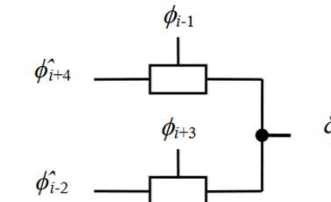
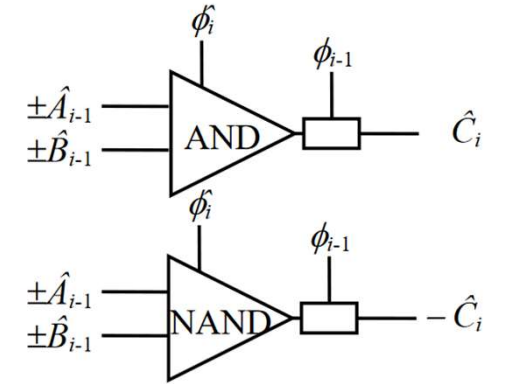
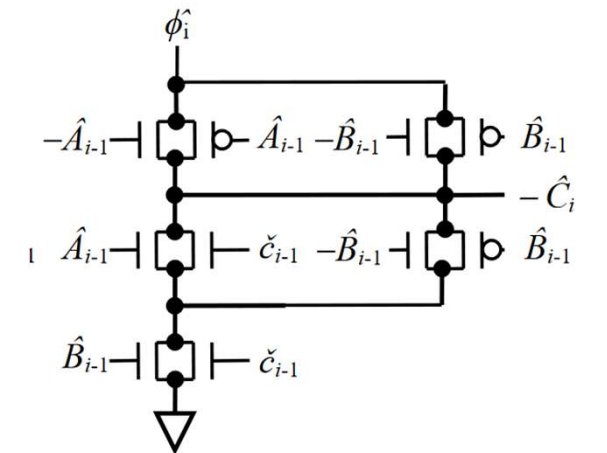



Fig. 4. (a) Unlatched adiabatic buffer from [4, Fig. 4], (b) same buffer for the negated signal, (c) however, the incoming cup signals can be generated from the negated signals in the previous stage, provided that a helper signal \check{c}_i is available. (d) The helper signal can be generated once in an entire circuit from available clocks.



(c) NAND, $-\hat{C}_i$ from [12, Fig. 9]





Reversible Computing Technologies in Superconducting Platforms

The Reversible Computing Future

Adiabatic Reversible Computing in Superconducting Circuits



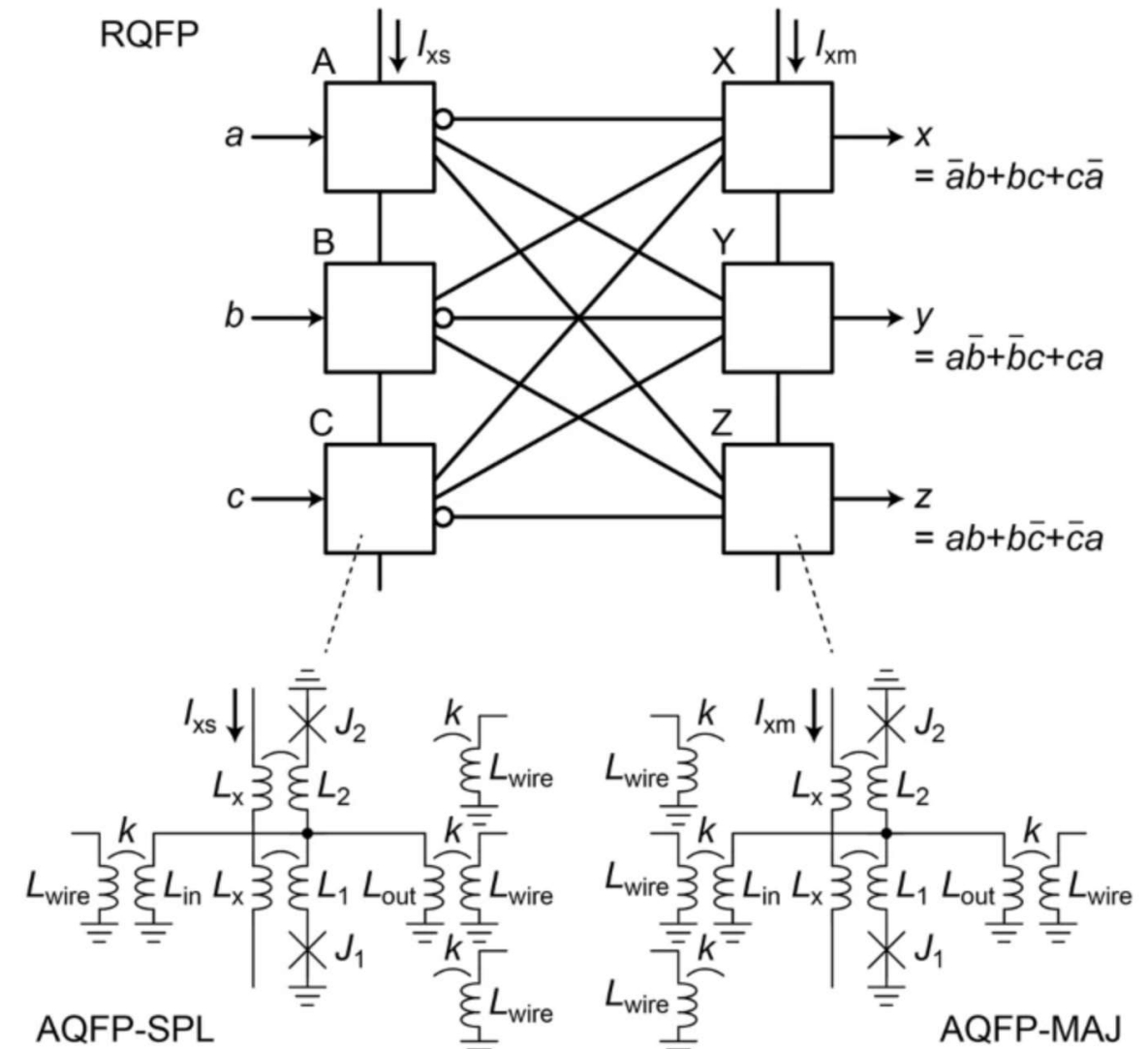
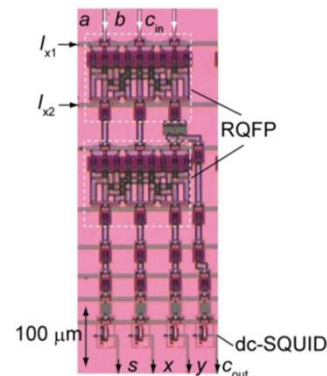
Work along this general line has roots that go all the way back to Likharev, 1977. (doi:10.1109/TMAG.1977.1059351)

- Most active group recently is Prof. Yoshikawa's group at Yokohama National University in Japan.

Logic style called *Reversible Quantum Flux Parametron* (RQFP).

- Shown at right is a 3-output *reversible majority gate*.
- Full adder circuits have also been built and tested.

Simulations indicate that RQFP circuits can dissipate $< kT \ln 2$ (even noting that $T = 4\text{K}$), at speeds on the order of 10 MHz



Existing Dissipation-Delay Products (DdP)— Adiabatic Reversible Superconducting Circuits

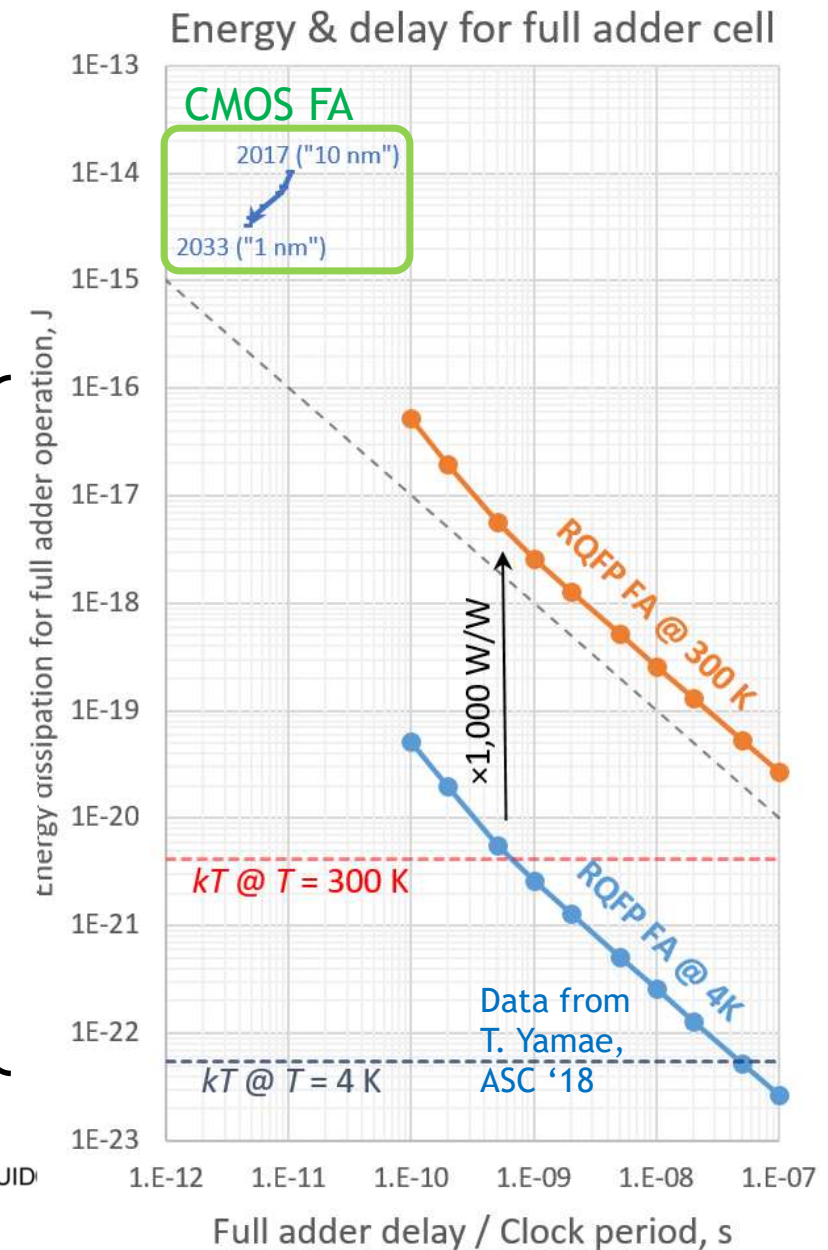
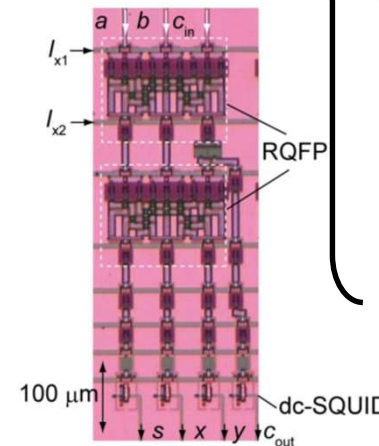
Reversible adiabatic superconductor logic:

- State-of-the-art is the **RQFP** (Reversible Quantum Flux Parametron) technology from Yokohama National University in Japan.
- Chips were fabricated, function validated.
- Circuit simulations predict DdP is $>1,000\times$ *lower* than even *end-of-roadmap* CMOS.
- Dissipation extends *far below* the 300K Landauer limit (and even below the Landauer limit at 4K).
- DdP is *still* better than CMOS even after adjusting by a conservative factor for large-scale cooling overhead (1,000 \times).

Question: Could some *other* reversible technology do even better than this?

- We have a project at Sandia exploring one possible superconductor-based approach for this (more later)...
- But, what are the *fundamental* (technology-independent) limits, if any?

RQFP =
Reversible
Quantum Flux
Parametron
(Yokohama U.)



Ballistic Reversible Computing

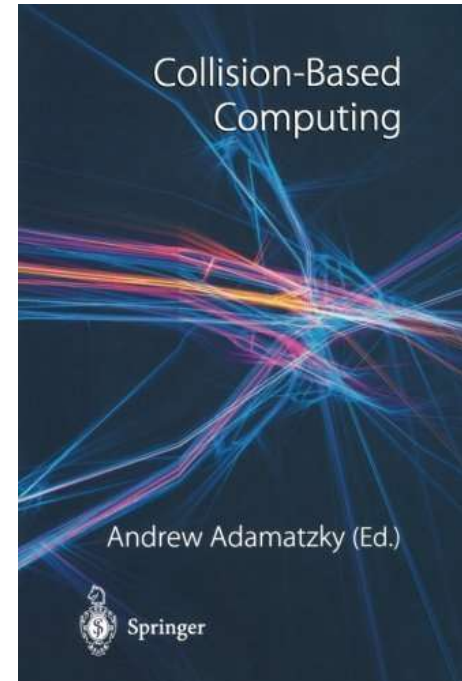
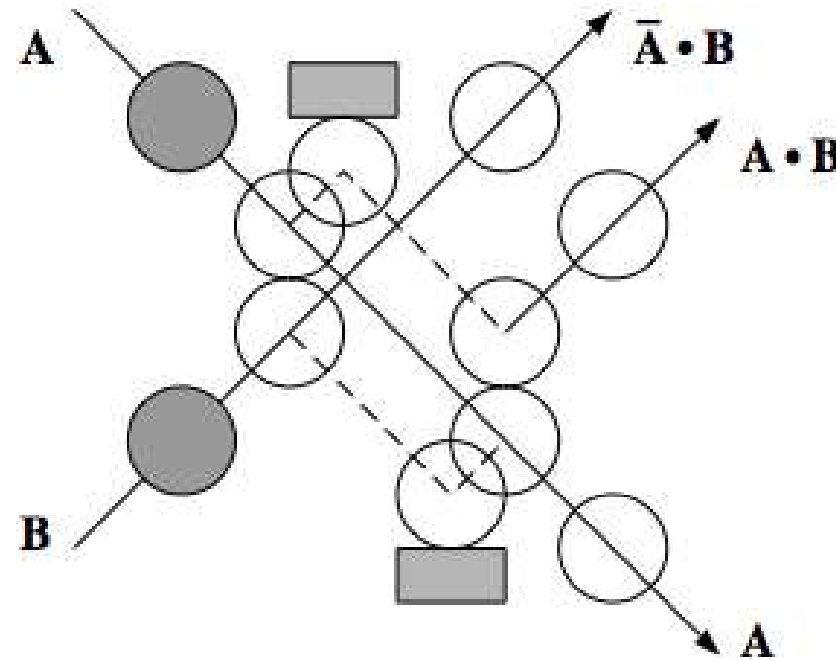
Can we envision reversible computing as a *deterministic* elastic interaction process?

Historical origin of this concept:

- Fredkin & Toffoli's *Billiard Ball Model* of computation ("Conservative Logic," IJTP 1982).
 - Based on elastic collisions between moving objects.
 - Spawned a subfield of "collision-based computing."
 - Using localized pulses/solitons in various media.

No power-clock driving signals needed!

- Devices operate when data signals arrive.
- The operation energy is carried by the signal itself.
 - Most of the signal energy is preserved in outgoing signals.



However, all (or almost all) of the existing design concepts for ballistic computing invoke implicitly *synchronized* arrivals of ballistically-propagating signals...

- Making that approach work in reality presents some serious difficulties, however:
 - Unrealistic in practice to assume precise alignment of signal arrival times.
 - Thermal fluctuations & quantum uncertainty, at minimum, are always present.
 - Any relative timing uncertainty leads to chaotic dynamics when signals interact.
 - Exponentially-increasing uncertainties in the dynamical trajectory.
 - Deliberate *resynchronization* of signals whose timing relationship has become uncertain incurs an inevitable energy cost.

Can we come up with a *new* ballistic model of reversible computing that avoids these problems?

Ballistic Asynchronous Reversible Computing (BARC)



Problem: Conservative (dissipationless) dynamical systems generally tend to exhibit chaotic behavior...

- This results from direct nonlinear *interactions* between multiple continuous dynamical degrees of freedom (DOFs), which amplify uncertainties, exponentially compounding them over time...
- *E.g.*, positions/velocities of ballistically-propagating “balls”
 - Or more generally, any localized, cohesive, momentum-bearing entity: Particles, pulses, quasiparticles, solitons...

Core insight: In principle, we can greatly reduce or eliminate this tendency towards dynamical chaos...

- We can do this simply by *avoiding* any direct interaction between continuous DOFs of different ballistically-propagating entities

Require localized pulses to arrive *asynchronously*—and furthermore, at clearly distinct, *non-overlapping* times

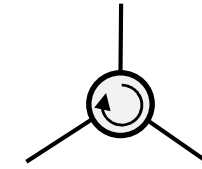
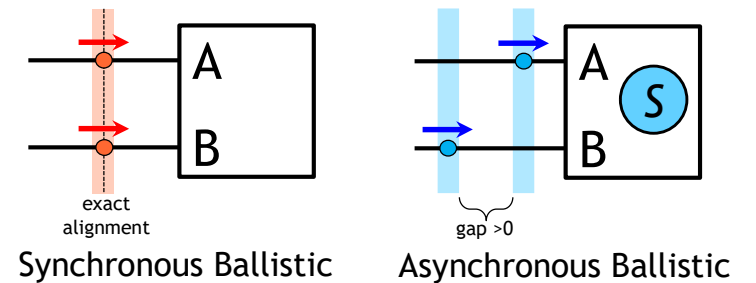
- Device’s dynamical trajectory then becomes *independent* of the precise (absolute *and* relative) pulse arrival times
 - As a result, timing uncertainty per logic stage can now accumulate only *linearly*, not exponentially!
 - Only relatively occasional re-synchronization will be needed
- For devices to still be capable of doing logic, they must now maintain an internal discrete (digitally-precise) state variable—a stable (or at least metastable) stationary state, *e.g.*, a ground state of a well

No power-clock signals, unlike in adiabatic designs!

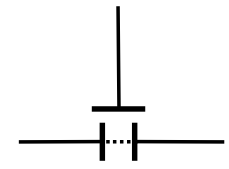
- Devices simply operate whenever data pulses arrive
- The operation energy is carried by the pulse itself
 - Most of the energy is preserved in outgoing pulses
 - Signal restoration can be carried out incrementally, or periodically

Goal of current effort at Sandia: Demonstrate BARC principles in an implementation based on fluxon dynamics in Superconducting Electronics (SCE)

(BARCS effort)

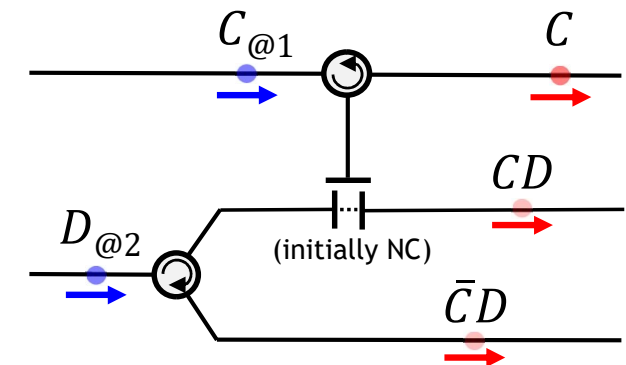


Rotary
(Circulator)



Toggled
Barrier

Example BARC device functions



Example logic construction

Simplest Fluxon-Based (bipolarized) BARC Function



One of our early tasks: Characterize the simplest nontrivial BARC device functionalities, given a few simple design constraints applying to an SCE-based implementation, such as:

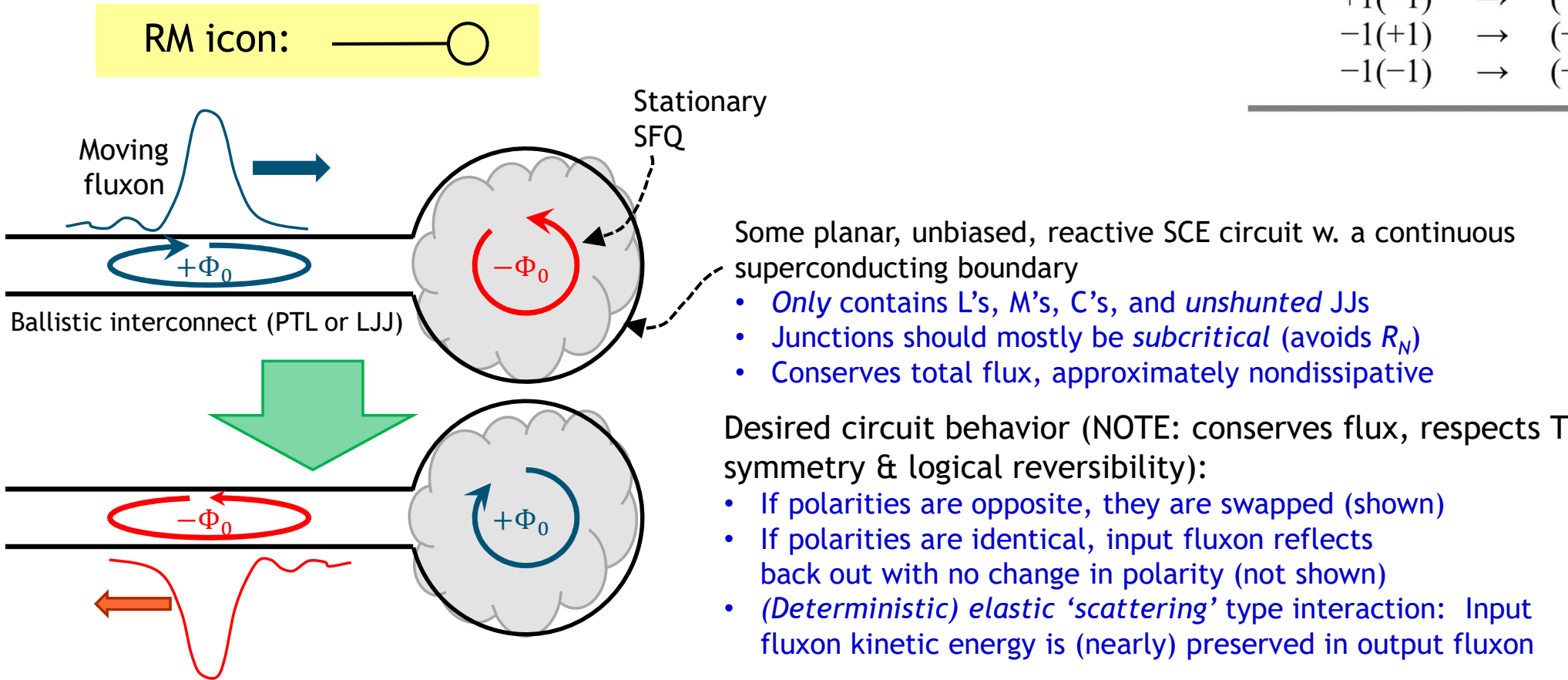
- (1) Bits encoded in fluxon polarity; (2) Bounded planar circuit conserving flux; (3) Physical symmetry.

Determined through theoretical hand-analysis that the simplest such function is the 1-Bit, 1-Port Reversible Memory Cell (RM):

- Due to its simplicity, this was then the preferred target for our subsequent detailed circuit design efforts...

RM Transition Table

Input Syndrome	Output Syndrome
+1(+1)	→ (+1)+1
+1(−1)	→ (+1)−1
−1(+1)	→ (−1)+1
−1(−1)	→ (−1)−1



RM—First working (in simulation) implementation!

Erik DeBenedictis: “Try just strapping a JJ across that loop.”

- This actually works!

“Entrance” JJ sized to = about 5 LJJ unit cells ($\sim 1/2$ pulse width)

- I first tried it twice as large, & the fluxons annihilated instead...
 - “If a $15 \mu\text{A}$ JJ rotates by 2π , maybe $1/2$ that will rotate by 4π ” 🤔

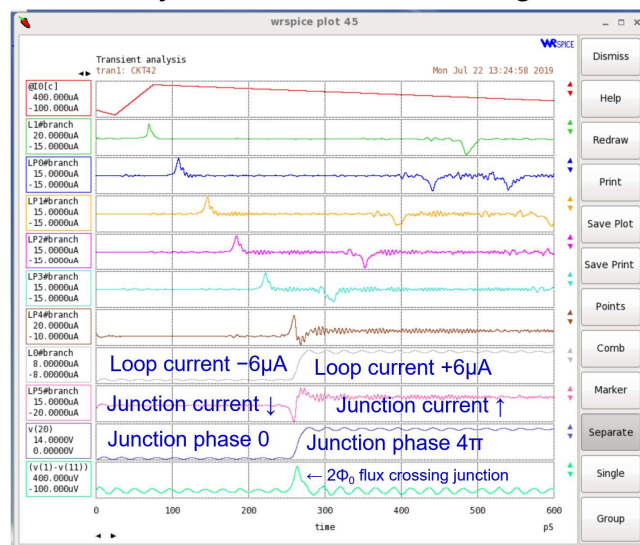
Loop inductor sized so ± 1 SFQ will fit in the loop (but not ± 2)

- JJ is sitting a bit below critical with ± 1

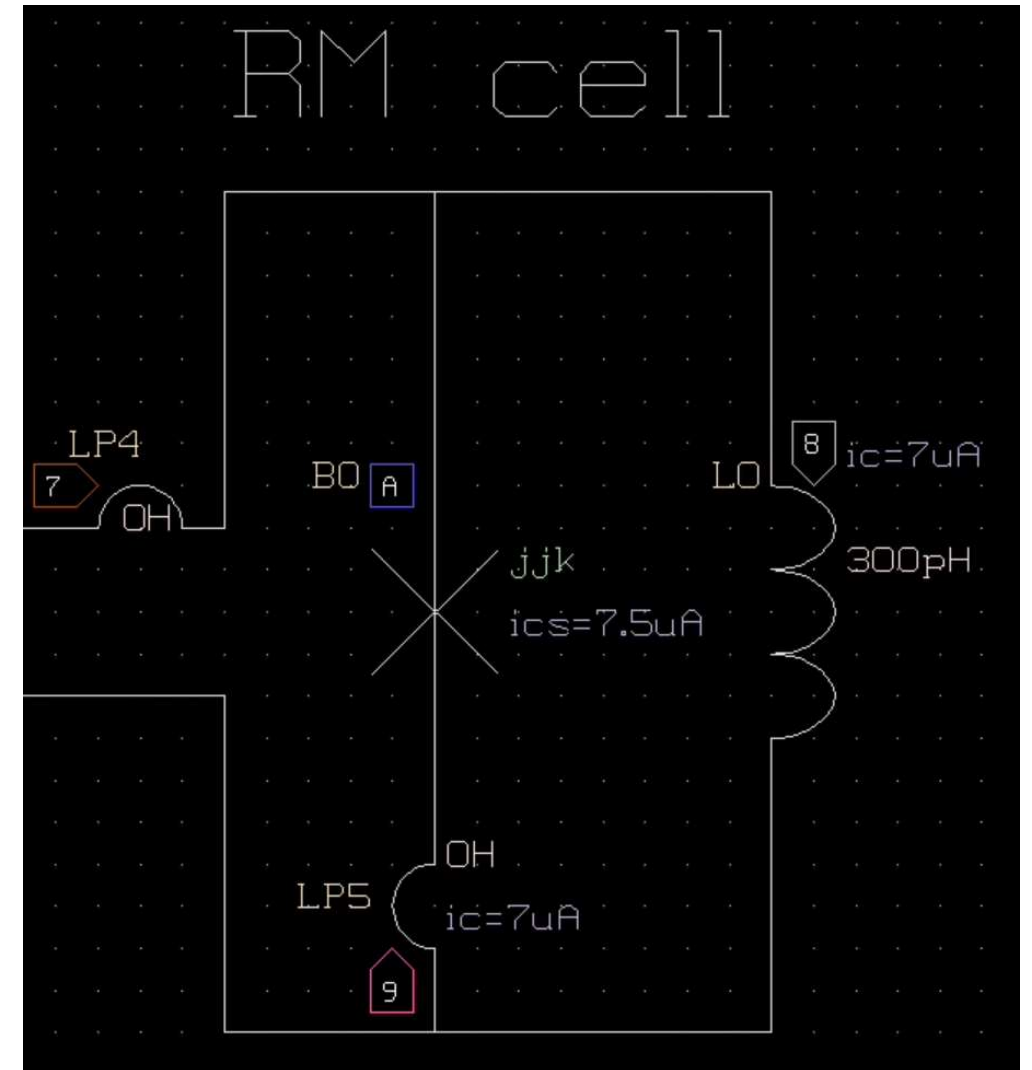
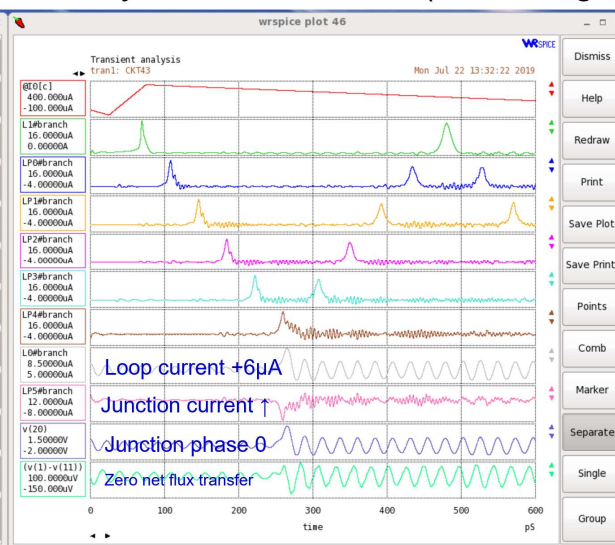
WRspice simulations with ± 1 fluxon initially in the loop

- Uses `ic` parameter, & `uic` option to `.tran` command
 - Produces initial ringing due to overly-constricted initial flux
 - Can damp w. small shunt G

Polarity mismatch \rightarrow Exchange



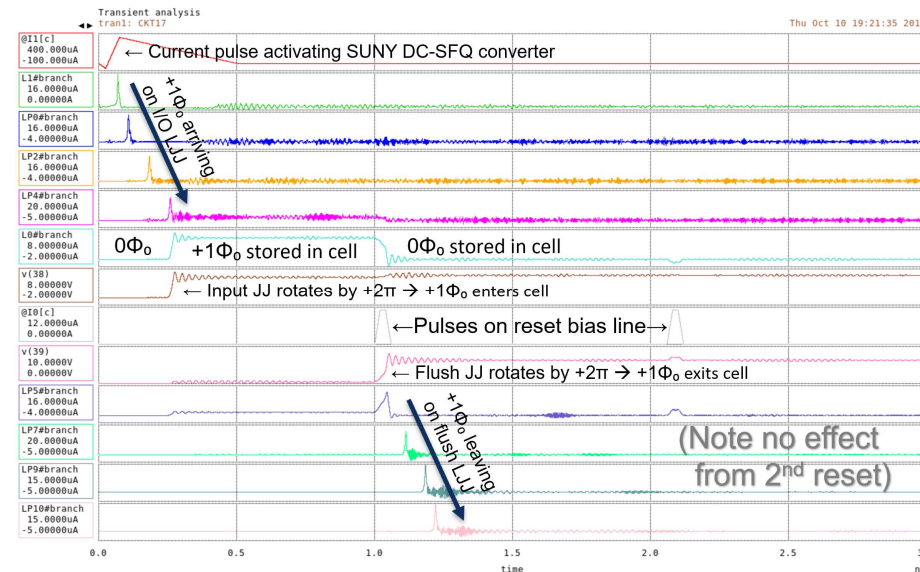
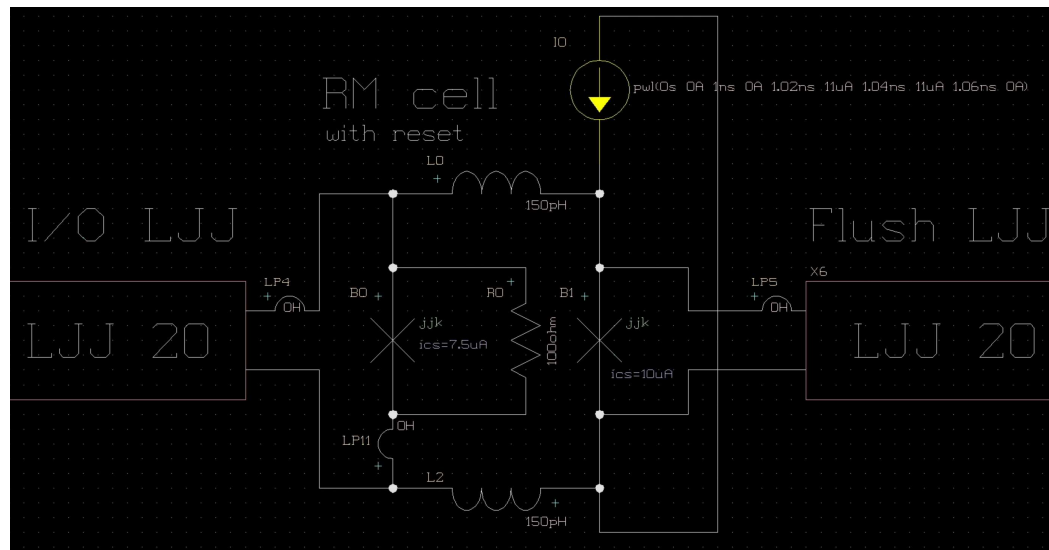
Polarity match \rightarrow Reflect (=Exchange)



Resettable version of RM cell—Designed & Fabricated!

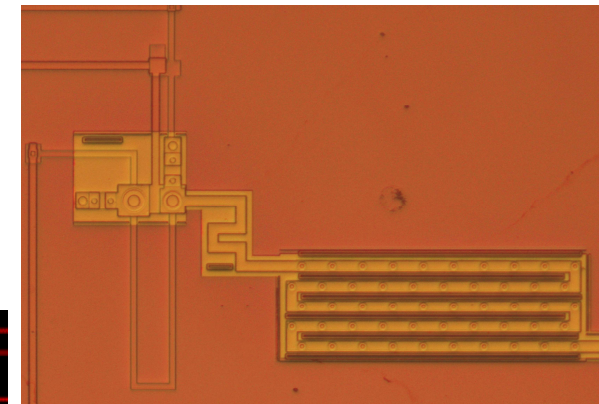
Apply current pulse of appropriate sign to flush the stored flux (the pulse here flushes out positive flux)

- To flush either polarity \rightarrow Do both (\pm) resets in succession

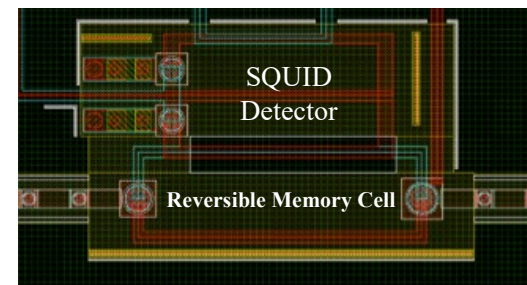
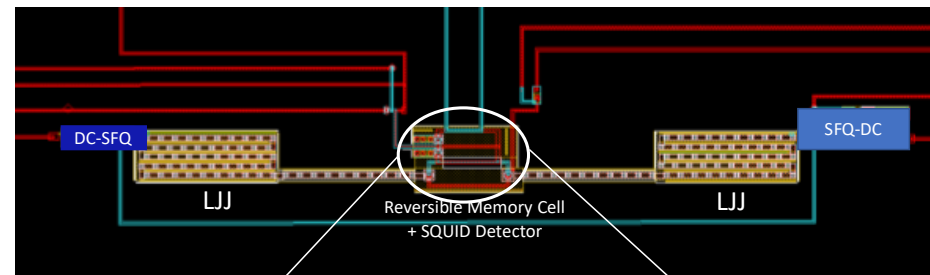
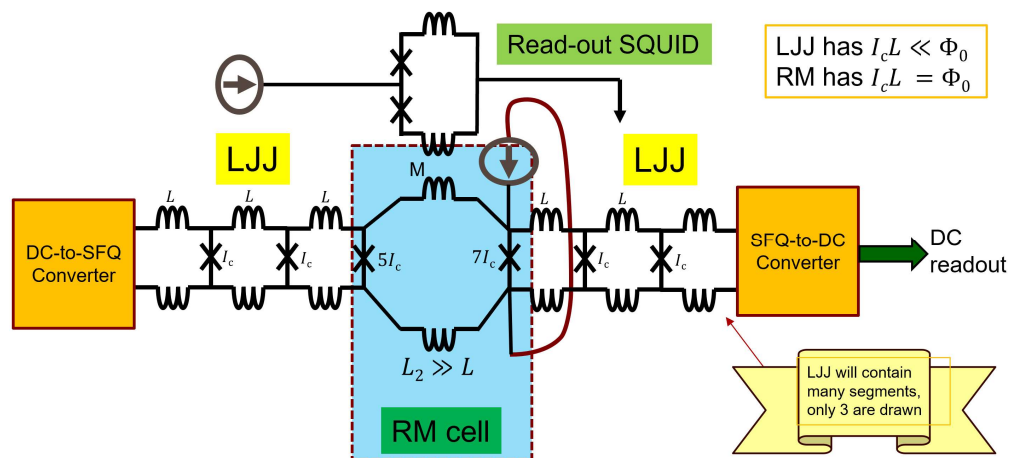
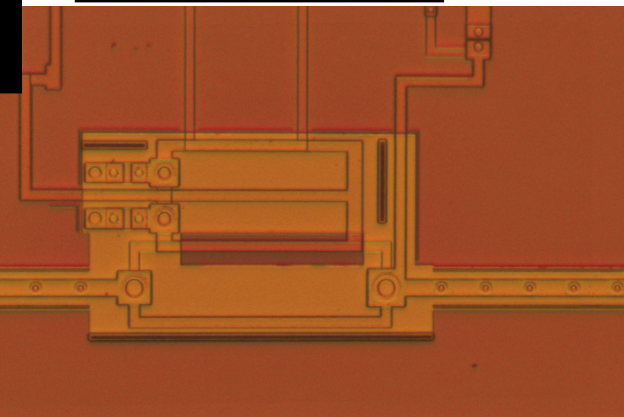


Fabrication at SeeQC with support from ACI

DC-SFQ & LJJ



RM Cell & SQUID





Physics/Engineering Challenges & Conclusion

The Reversible Computing Future

R&D To Do for Reversible Computing



To improve the (potential) efficiency of existing CMOS-based technology platforms for RC:

- Continue development of high- Q trapezoidal resonators, optimize packaging & integration.
- Re-engineer FET device structures to (more) aggressively minimize leakage.
- Improve cost-efficiency of 3D fabrication processes.

To improve the energy-delay product of RC implementations (across a range of temperatures):

- Need to identify practical RC devices leveraging “exotic” (quantum-mechanics-based) operating principles.
- Work is needed to characterize the fundamental limits of efficiency of RC as a function of various physical timescales of interest.
 - *E.g.*, equilibration, relaxation, fluctuation, decoherence, and switching/interaction timescales are (potentially) all important.

To develop practical digital circuits & systems based on RC, we need:

- Extensions to EDA tools are needed to support reversible circuits & architectures.
- New RC-based hardware designs (hardware algorithms for functional units, IP blocks, processor designs).
- (Eventually) reversible programming models/languages & software algorithms.
 - There is substantial work in this area already.

Conclusion

There will never *not* be a pressing demand for ever more-efficient general digital computing!

- This will remain true *despite* the emergence of a variety of non-digital computing models (e.g., analog, dynamical-systems based, stochastic, quantum) for various specialized applications.

The conventional (non-reversible) paradigm for digital computing is approaching its end-of-life.

- Soon it will no longer be possible to improve its efficiency due to fundamental thermodynamic limits.

Reversible computing offers the *only* physically possible route to continue improving the efficiency of digital computing beyond the limits of the non-reversible paradigm.

- And further, we know of no fundamental limits to the energy-efficiency (and cost-efficiency!) of RC.

Various groups have already demonstrated clear, compelling proofs-of-concept for the implementation of RC in both semiconducting and superconducting technology platforms.

- At this point, there really is nothing fundamental that prevents the further development of RC technology towards eventual commercialization.

Of course, much work remains to be done if we wish to continue improving the efficiency and scale of RC, but no *fundamental* barriers to further ongoing improvement are apparent.

- \therefore RC is a nascent new subfield of ECE that is now quite ripe for significant further development.

Really, the only thing needed at this point is simply **massive levels of new R&D funding** (from government, industry, &/or far-sighted investors).

- IMO, we really need dedicated funding to ramp up to a level of (at least) \$100M/year in order to make an adequately rapid rate of R&D progress across the entire field if we want to have solutions ready to go by the time the efficiency of non-reversible digital technology totally flatlines...

RC could grow the value of the digital economy by many orders of magnitude.

